

SECURE DATA TRANSMISSION SCHEME FOR INTERNET OF THINGS

BY
IBRAHEEM MOHAMMED AHMED AL-HEJRI

A Dissertation Presented to the
COLLEGE OF GRADUATE AND INTERDISCIPLINARY STUDIES
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY
In
COMPUTER SCIENCE

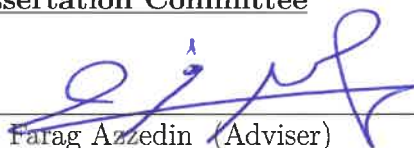
DECEMBER, 2023

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This dissertation, written by IBRAHEEM MOHAMMED AHMED AL-HEJRI under the direction of his dissertation adviser and approved by his dissertation committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **DOCTOR OF PHILOSOPHY IN INFORMATION AND COMPUTER SCIENCE**.

Dissertation Committee


Dr. Farag Azzedin (Adviser)


Dr. Sultan Al-Muhammadi (Co-Adviser)


Dr. Tarek Sheltami (Member)


Dr. Mohammad Alshayeb (Member)


Dr. Ahcene Bounceur (Member)



Dr. Motaz Alfarraj
Department Chairman


Dr. Suliman Al-Homidan
Dean of Graduate Studies



Date

©Ibraheem Mohammed Ahmed Al-Hejri
2023

*With all of my love to my mother's soul, my father, my wife and
my brothers*

ACKNOWLEDGMENTS

First and foremost, I would like to express my deepest gratitude to Allah Almighty as with the blessing this project has successfully been concluded. I would like to extend my heartfelt gratitude to my advisor Dr. Farag Azzedin for his unwavering dedication, extensive time investment, invaluable guidance, and patience throughout the entirety of this thesis. His tireless commitment to my academic journey has transcended the role of a mere supervisor, as he has served as my mentor, teacher, and even a father figure. I consider myself incredibly fortunate to have had him as my thesis supervisor. I would also like to thank my co-adviser Dr. Sultan Almuhammadi for his constant support. He has always been there to listen and answer my questions, read and review my report. I also thank my thesis committee members: Dr. Tarek Sheltami, Dr. Mohammad Alshayeb, and Dr. Ahcene Bounceur for their insightful comments and advice on my thesis and final defense. I am also grateful to all my family members for their continuous support. Also, to my dear wife who has been a constant source of inspiration with her enthusiasm and positivity. Thanks for all their patience and taking all the pain to give me time to finish this work. Special thanks to Information and Computer Science Department at KFUPM for their invaluable support.

Contents

ACKNOWLEDGMENTS	vi
LIST OF TABLES	xi
LIST OF FIGURES	xiii
LIST OF ABBREVIATIONS	xv
ABSTRACT (ENGLISH)	xvii
ABSTRACT (ARABIC)	xix
CHAPTER 1 INTRODUCTION	1
1.1 Problem Statement and Motivations	4
1.2 Thesis Main Contributions	7
1.3 Research Methodology	8
1.4 Thesis Organization	9
CHAPTER 2 BACKGROUND	11
2.1 IoT Security Attacks	12
2.2 Application Layer Attacks	14
2.3 Transportation Layer Attacks	17
2.4 Perception Layer Attacks	22
2.5 Attacks Based on Access Level	27
2.6 Attacks Based on Geographic Location	28
2.7 Attacks Based on Data Damage Level	29
2.8 Attacks Based on Objects of Vulnerability	31

2.9	IoT Security Attacks Taxonomy	32
CHAPTER 3 INTERNET-OF-THINGS DATA SECURITY		34
3.1	Overview	35
3.2	Data States Security Challenges	36
3.3	Data at Rest Countermeasures	40
3.4	Data in Use Countermeasures	42
3.5	Data in Transit Countermeasures	45
3.6	Countermeasures Taxonomy	50
3.7	Concluding Remarks	53
3.8	Future Directions	54
CHAPTER 4 CRYPTOGRAPHIC TECHNIQUES		56
4.1	Overview	57
4.2	Symmetric Cryptography	59
4.2.1	Advanced Encryption Standard	60
4.2.2	Limitations of Symmetric Key Cryptography	61
4.3	Public Key Cryptography	63
4.3.1	Elliptic Curve Cryptography	64
4.3.2	Elliptic Curve Menezes-Qu-Vanstone	66
4.3.3	Limitations of Asymmetric Key Cryptography	67
CHAPTER 5 LITERATURE REVIEW		69
5.1	Non-ECC Based Schemes	69
5.2	ECC Based Schemes	72
5.3	Lightweight ECC-Based Schemes Limitations	76
CHAPTER 6 PROPOSED LIGHTWEIGHT SECURE DATA TRANSMISSION SCHEME		78
6.1	LSDT Architecture	79
6.2	LSDT Overview	81
6.3	LSDT Phases	83

6.3.1	Initialization Phase	84
6.3.2	Key Establishment Phase Between CM and CH	85
6.3.3	Key Establishment Phase Between BS and CH	86
6.3.4	Data Transmission Phase	88
6.4	The Trade-offs of LSDT Scheme	90
6.5	LSDT Applications in Wireless Networks	92
CHAPTER 7 PERFORMANCE AND SECURITY ANALYSIS		95
7.1	Performance Analysis	95
7.1.1	Communication Cost	95
7.1.2	Computation Cost	97
7.2	Security Analysis	98
7.3	Scalability and Robustness of LSDT	100
7.3.1	LSDT Scalability	100
7.3.2	LSDT Robustness	103
7.3.3	Scalability Limitations	103
7.4	Comparative Analysis	105
7.4.1	Security Comparative Analysis	105
7.4.2	Performance Comparative Analysis	106
CHAPTER 8 PERFORMANCE EVALUATION		109
8.1	Simulation Environment	109
8.1.1	Contiki Operating System	110
8.1.2	Cooja Simulator	111
8.2	Experimental Methodology	113
8.3	Simulation Results and Discussion	114
CHAPTER 9 CONCLUSION AND FUTURE WORK		126
9.1	Conclusions	127
9.1.1	IoT Data Security Survey	127

9.1.2	Lightweight Secure Data Transmission Scheme for IoT En- vironments	129
9.2	Limitations	130
9.3	Future Directions	131
9.4	Thesis Outcomes	132
REFERENCES		134
VITAE		172

List of Tables

Table 2.1	Key Differences between Conventional and IoT Security . . .	13
Table 2.2	Summary of Main Existing Attacks in IoT Model	27
Table 2.3	Summary of Existing IoT Attacks	33
Table 3.1	Challenges of Data Security in IoT	39
Table 3.2	Summary of Existing Studies for Data at Rest	43
Table 3.3	Summary of Existing Studies for Data in Use	44
Table 3.4	Summary of Existing Studies for Data in Transit	51
Table 5.1	Limitations of Existing Non-ECC Based Schemes	71
Table 5.2	Summary of Existing ECC-Based Schemes	76
Table 5.3	Limitations of Existing Lightweight ECC-Based Schemes . . .	77
Table 6.1	Proposed Scheme Notations	81
Table 7.1	LSDT Communication Cost	97
Table 7.2	LSDT Computation Cost	98
Table 7.3	Comparative Study on Security Services	105
Table 7.4	Security Services Comparison	106
Table 7.5	Comparative Analysis of Communication Costs	107
Table 7.6	Comparative Analysis of Computation Costs	108
Table 8.1	Simulation Configuration	112
Table 8.2	LSDT Computational Time	116
Table 8.3	Comparison of Communication Cost	122
Table 8.4	Comparison of Computation Cost	123
Table 8.5	Comparative Communication Cost Analysis	124

Table 8.6 Comparative Computation Cost Analysis 125

List of Figures

Figure 2.1	IoT Design Model	13
Figure 2.2	IoT Security Attacks	32
Figure 3.1	The Three Data States	37
Figure 3.2	Taxonomy of Countermeasures Based on The Three Data States.	52
Figure 4.1	AES Round Structure	61
Figure 4.2	Point Addition on ECC [1]	65
Figure 4.3	ECMQV Key Agreement Protocol	67
Figure 6.1	Network Architecture	80
Figure 6.2	Proposed Scheme Phases	84
Figure 6.3	Key Establishment (CH-BS) Phase Flow Chart	87
Figure 6.4	Data Transmission Phase Flow Chart	89
Figure 8.1	Experimental Process Approach	113
Figure 8.2	Network Topology	114
Figure 8.3	Communication Cost - Key Establishment Phase	116
Figure 8.4	Communication Cost - Data Transmission Phase	117
Figure 8.5	Total Communication Cost Comparison	118
Figure 8.6	Computation Cost - Key Establishment Phase	119
Figure 8.7	Computation Cost - Data Transmission Phase	120
Figure 8.8	Total Computation Cost Comparison	121
Figure 8.9	Comparison of Communication Cost	122
Figure 8.10	Comparison of Computation Cost	123
Figure 8.11	Comparative Communication Cost Analysis	124

Figure 8.12 Comparative Computation Cost Analysis	125
---	-----

LIST OF ABBREVIATIONS

AES	Advanced Encryption Standard
APT	Advanced Persistent Threat
BDD	Binary Decision Diagrams
BS	Base Station
CH	Cluster Head
CM	Cluster Member
DiTD	Innovative Data-in-Transit Defender
DLP	Data Leakage Prevention
DoS	Denial-of-Service
DSA	Digital Signature Algorithm
ECC	Elliptic Curve Cryptography
ECDH	Elliptic-Curve Diffie-Hellman
ECDLP	Elliptic Curve Discrete Logarithmic Problem
ECMQV	Elliptic Curve Menezes-Qu-Vanstone
FHE	Fully Homomorphic Encryption
GPS	Global Positioning System

IoT	Internet of Things
LSDT	Lightweight Secure Data Transmission
MAC	Message Authentication Code
MCI	Malicious Code Injection
MITM	Man-In-The-Middle
MPM	Multiple Point Multiplication
PUFs	Physical Unclonable Functions
RFID	Radio-Frequency IDentification
RSA	Rivest-Shamir-Adleman
RSN	RFID Sensor Network
SDTS	Secure Data Transmission Scheme
WSNs	Wireless Sensor Networks

DISSERTATION ABSTRACT

NAME: Ibraheem Mohammed Ahmed Al-Hejri
TITLE OF STUDY: Secure Data Transmission Scheme for Internet of Things
MAJOR FIELD: Computer Science
DATE OF DEGREE: December, 2023

The Internet of Things (IoT) is rapidly permeating critical domains, enabling the interconnection and utilization of diverse devices on a large scale while streaming vast volumes of data. In domains like telehealth, intelligent transportation, and autonomous agriculture, ensuring the confidentiality, integrity, and authenticity of collected and exchanged data is paramount. However, the resource limitations and heterogeneous nature of IoT devices often render traditional cryptography-based techniques ineffective or even infeasible for secure data transmission. Consequently, there is an urgent need to develop and implement lightweight, secure, and scalable schemes for data transmission. In this thesis, a novel Lightweight Secure Data Transmission (LSDT) scheme for IoT environments is proposed. LSDT scheme is composed of three phases and utilizes an efficient combination of sym-

metric keys and an elliptic-curve Menezes-Qu-Vanstone asymmetric key agreement protocol. Security and performance analysis indicate that the LSDT scheme exhibits robust resistance to various security attacks while maintaining the highest level of confidentiality, integrity, and end-to-end mutual authentication. Furthermore, we demonstrate the effectiveness and efficiency of the proposed LSDT scheme through simulation experiments and comparative analysis with related security schemes. The results of the comparison reveal that LSDT outperforms other relevant schemes in terms of communication and computation costs, making it a highly suitable and efficient option for IoT applications.

ملخص الرسالة

الاسم: إبراهيم محمد أحمد الهجري

عنوان الدراسة: نظام نقل البيانات الآمن لإنترنت الأشياء

التخصص: علوم الحاسب الآلي

تاريخ الدرجة العلمية: جمادى الآخرة 1445

تتغلغل إنترنت الأشياء (IoT) بسرعة في المجالات الحيوية، مما يتيح التوصيل البيئي واستخدام الأجهزة المتنوعة على نطاق واسع مع تدفق كميات هائلة من البيانات. وفي مجالات عديدة مثل الرعاية الصحية عن بعد، والنقل الذكي، والزراعة الذاتية، يعد ضمان سرية وسلامة وصحة البيانات التي يتم جمعها وتبادلها أمراً بالغ الأهمية. ومع ذلك، فإن القيود المفروضة على الموارد والطبيعة غير المتجانسة لأجهزة إنترنت الأشياء غالباً ما تجعل التقنيات التقليدية القائمة على التشفير غير فعالة أو حتى غير مجدية لنقل البيانات بشكل آمن. وبالتالي، هناك حاجة ملحة لتطوير وتنفيذ أنظمة وخطط خفيفة الوزن وأمنة وقابلة للتطوير لنقل البيانات. في هذه الأطروحة، تم اقتراح نظام جديد خفيف الوزن و آمن لنقل البيانات (LSDT) لبيئات إنترنت الأشياء. يتكون نظام LSDT من ثلاث مراحل ويستخدم مزيجاً فعالاً من المفاتيح المتماثلة وبروتوكول اتفاقية المفاتيح غير المتماثلة ذات المنحنى البيضواوي Menezes-Qu-Vanstone. يشير تحليل الأمان والأداء إلى أن نظام LSDT يُظهر مقاومة قوية لمختلف الهجمات الأمنية مع الحفاظ على أقصى درجات السرية والنزاهة والمصادقة المتبادلة الشاملة. علاوة على ذلك، قمنا بإظهار فعالية وكفاءة نظام LSDT المقترح من خلال تجارب المحاكاة والتحليل المقارن مع الأنظمة الأمنية ذات الصلة. تكشف نتائج المقارنة أن LSDT يتفوق على الأنظمة و المخططات الأخرى ذات الصلة من حيث تكاليف الاتصالات والحوسبة، مما يجعله مناسباً وفعالاً للغاية لتطبيقات إنترنت الأشياء.

CHAPTER 1

INTRODUCTION

In recent years, the emergence of the Internet of Things (IoT) has led to a paradigm change in all aspects of human-machine interaction [2]. IoT enables various physical objects and methods of communication to achieve a certain task by exchanging information [3]. IoT exploits underlying technologies to make these objects much smarter such as Wireless Sensor Networks (WSNs), Internet protocols and embedded devices [4], [5]. IoT trend is gaining popularity and is expected to be the most significant element in the coming decades that will influence fundamental business logic [6].

In face of this rapid change, data security has brought critical challenges in designing and implementing IoT applications. Since the IoT is primarily dependent on the Internet infrastructure, it opens itself up to all conventional and cybersecurity threats. Cybersecurity attacks on IoT not only endanger IoT functionality but also expose human lives to risks due to security and privacy breaches [7].

Furthermore, critical infrastructures such as oil and gas supply, healthcare systems, and power grids face significant security challenges in today's interconnected

world. One of the major concerns is the increasing sophistication and frequency of cyber attacks. These infrastructures heavily rely on computerized systems and interconnected networks, making them vulnerable to malicious actors aiming to steal sensitive data, disrupt operations, or cause physical damage [8], [9]. In the oil and gas industry, for example, cyber attacks can disrupt the production and distribution of energy resources, leading to significant economic and environmental consequences [10], [11].

Meeting the reliability and security requirements for these critical infrastructures is of paramount importance. A reliable infrastructure ensures the continuous operation of essential services, safeguards public safety, and protects economic stability. It requires a multi-faceted approach that includes robust cybersecurity measures, regular vulnerability assessments, and proactive incident response strategies. By implementing stringent security protocols, such as access controls and encryption, critical infrastructures can strengthen their defenses against cyber threats [8], [12], [13].

Reducing the overhead and complexity of security defenses for critical infrastructures is crucial in effectively mitigating cyber threats. Simplifying security measures helps streamline operations, improve efficiency, and enhance the overall effectiveness of defensive strategies. Complex security defenses can introduce unnecessary overhead, making it challenging to manage and maintain the security posture of the infrastructure. This complexity often leads to an increased risk of misconfigurations, vulnerabilities, and human errors, which can be exploited by cyber attackers [14]. Moreover, reducing overhead and complexity allows for

faster response times in the event of a security incident [15], [16]. Simplified security measures enable security teams to identify and respond to threats more efficiently, reducing the time required for analysis, investigation, and remediation. This agility is crucial in critical infrastructures where any delay in addressing a cyber threat can have severe consequences, such as prolonged downtime, financial losses, or compromised public safety [17]. Overall, by streamlining security defenses, critical infrastructures can enhance their resilience, reduce operational costs, and enable rapid response to cyber threats. A simplified and efficient security framework allows organizations to focus on proactive threat intelligence, continuous monitoring, and timely incident response, thereby strengthening their ability to protect essential services and maintain the reliability and security of critical infrastructures [18].

The current trend in IoT security is towards providing lightweight and efficient schemes to secure data transmission. Ensuring the security of IoT applications is a challenge due to the inability of a considerable proportion of resource-limited IoT devices to implement proper security schemes [19]–[21]. Even though tremendous security defenses and countermeasures have been developed, these solutions cannot be applied directly to IoT infrastructures due to the major difference in the computation and storage capabilities between conventional computing and IoT devices [22]–[24].

According to the 2020 conceptual framework [25], data is the key component in IoT systems [26], [27]. In IoT environment, the data is created outside of the data center and therefore the data creation, computing, and management can be

performed outside of the core data center. This challenge places the responsibility on IoT devices to perform heavy computational tasks and may even cause a large delay on service response [27]–[29]. The data in IoT systems can be classified into three different states: data at rest, data in use, and data in transit. The main concern for a large proportion of IoT applications is having a secure environment to ensure the secrecy of the data. Over recent years, the main focus for many researchers has been on how to securely transport security-sensitive data between IoT entities. As such, designing a lightweight, secure data transmission scheme for IoT environments is becoming essential [30], [31].

1.1 Problem Statement and Motivations

Over the past decades, significant advancements have been made in developing security defenses and countermeasures for traditional computation and communication environments. However, applying these solutions in the context of IoT presents unique challenges due to the substantial differences in computation, storage, and communication capabilities between conventional and IoT devices and functionalities [22]–[24]. As a result, there is a growing need for lightweight and efficient schemes to ensure secure data transmission in IoT environments.

This thesis focuses on securing data in transit, which is particularly vulnerable to attacks compared to other stages in the data life cycle [19]–[21], [32]–[34]. It is imperative to protect transmitted data from unauthorized access and various threats [19]–[21], [35]–[38]. Not only should the transmitted data be au-

thentic, but it should also remain free from unauthorized modifications [39]–[42]. Without achieving confidentiality, integrity, and authentication, the transmitted data becomes suspicious and may lead to undesirable consequences [30], [31], [43], [44]. Additionally, the diversity of communication channels, protocols, and techniques in IoT introduces further challenges for implementing effective security solutions [45]–[47].

The security of IoT data transmission has garnered significant attention [39]–[42], and various schemes have been proposed to ensure data transmission secrecy [48]–[53]. Given the paramount importance of securely transmitting sensitive data between IoT entities [33], [34], [54], [55], these schemes have implemented countermeasures to: (a) fulfill security requirements such as integrity [49] and mutual authentication proprieties [49], [50], and (b) mitigate attacks like Man-In-The-Middle (MITM) [49]–[51], replay, and brute force [48], [50], [51]. However, many of the proposed schemes suffer from high computation and communication overhead [48]–[52].

To address these critical challenges in secure data transmission within IoT environments, this thesis introduces a novel Lightweight Secure Data Transmission (LSDT) scheme. LSDT scheme utilizes an efficient combination of symmetric keys and the advanced Elliptic Curve Menezes-Qu-Vanstone asymmetric key agreement protocol [56]. Remarkably, LSDT exhibits lightweight characteristics, minimizing the communication and computation costs associated with securing data transmission. By ensuring confidentiality, integrity, and end-to-end mutual authentication, LSDT establishes a robust security framework that effectively safeguards against

various threats, including MITM attacks, brute force attacks, and replay attacks. Furthermore, this thesis provides an extensive analysis of LSDT's communication and computation costs, comparing them with other relevant security schemes, as detailed in Chapter 8. This comprehensive analysis enables a thorough evaluation of LSDT's performance and validates its superiority in terms of security and performance. By addressing these critical challenges and offering a lightweight and efficient solution, this thesis contributes significantly to the field of secure data transmission in IoT environments.

The proposed LSDT scheme is versatile and can be applied across various industries within the WSN-based IoT domains including e-healthcare systems, smart cities, and critical infrastructure systems like oil and gas supply, and power grids. The LSDT scheme addresses the crucial aspects of data security and authentication, which are integral to the seamless integration between IoT devices. Ensuring the security and integrity of data transmitted between IoT devices is of paramount importance in industries relying on IoT technology. By implementing LSDT scheme, organizations across diverse sectors can enhance the protection of sensitive data, establish secure communication channels, and authenticate the devices connected to the IoT network. For example, by applying LSDT scheme in e-healthcare systems, patients can establish secure and trustworthy communication channels with authorized doctors. Moreover, this scheme ensures the protection and confidentiality of all information exchanged between the patients and doctors, enhancing the overall security of the system. In conclusion, LSDT scheme offers a robust and adaptable solution that can be tailored to meet the spe-

cific security requirements of different IoT industries, facilitating their successful integration and operation.

1.2 Thesis Main Contributions

The dissertation contributions are detailed as follows:

1. Conducting a comprehensive survey for IoT security from the data security perspective with focus on attacks, challenges, and mitigation strategies of the data life cycle stages.
2. Proposing a lightweight secure data transmission (LSDT) scheme with the following features:
 - (a) Achieving low communication cost
 - (b) Achieving low computation cost
 - (c) Achieving the confidentiality property
 - (d) Achieving the data integrity property
 - (e) Achieving the end-to-end mutual authentication property
 - (f) Achieving the resistance to MITM, replay, and brute force attacks.
3. Developing a mathematical model for the proposed lightweight data transmission scheme.
4. Designing simulation experiments to evaluate the performance of the proposed scheme using IoT simulation tools.

1.3 Research Methodology

The following tasks are the method's steps which are followed to accomplish this work:

- **Task One : Survey**

Conduct a comprehensive survey covering the recently published works that have been done on IoT data security and the challenges, attacks and mitigation techniques of the data life cycle states. Then, investigate the security techniques that should be considered in the proposed scheme.

- **Task Two : Secure Data Transmission Scheme**

Design a lightweight secure data transmission scheme for IoT environments. The proposed scheme is lightweight (in terms of communication and computation costs) and provides secure data transmission that achieves confidentiality, integrity and end-to-end mutual authentication. In additions, this scheme resists MITM, brute force, and replay attacks.

- **Task Three : Mathematical Model**

Develop a mathematical model and use it to evaluate the performance of the proposed scheme in terms of communication and computation costs.

- **Task Four : Programming Skills**

Comprehend the programming of the platform and simulation tool used for the evaluation.

- **Task Five : Evaluation**

Through a comprehensive analysis and simulation experiments, evaluate the performance of the proposed scheme in terms of communication and computational costs. Then, the evaluation results are compared with the evaluation results in the relevant security schemes.

- **Task Six :** Writing up the dissertation.

1.4 Thesis Organization

The remainder of this thesis is divided into eight chapters as follows. **Chapte 2** discusses the potential IoT security attacks based on different aspects namely, architecture, location and access level of the attacks, data damage level, and objects of vulnerability. The taxonomy of IoT security attacks is also highlighted. **Chap-ter 3** elaborates on the need for data security and emphasizes its importance for organizations. This chapter also discusses the stages of data security with their challenges and countermeasures especially in IoT environments. In addition, this chapter highlights the taxonomy of IoT countermeasures. **Chapter 4** presents the basic concepts and preliminaries related to the modern cryptographic techniques such as Elliptic Curve Cryptography (ECC), Elliptic Curve Menezes-Qu-Vanstone (ECMQV) and Advanced Encryption Standard (AES). **Chapter 5** provides overview of efforts documented in various literature for different lightweight data transmission schemes have been put forth with the objective of fortifying the security of data transmission within IoT environments. **Chapter 6** describes

the proposed LSDT scheme architecture and phases. In addition, this chapter discusses the trade-offs made as a result of the proposed scheme in different aspects, namely security, performance, resource utilization, and energy efficiency. Furthermore, this chapter presents the proposed scheme applications in wireless networks. **Chapter 7** discusses the performance and security analyses of the proposed LSDT scheme. In addition, this chapter discusses the scalability and robustness of the proposed LSDT scheme. Furthermore, this chapter presents the comparative analysis of LSDT scheme with other relevant security schemes. **Chapter 8** presents the performance evaluation of the proposed LSDT scheme in terms of communication and computation costs. This chapter also provides detailed information on the construction of the experimental platform and outlines the approach taken in conducting our experiments. Furthermore, this chapter discusses the experiment results and compares the performance of LSDT scheme with other relevant security schemes in terms of communication and computation costs. **Chapter 9** concludes the work of this thesis and outlines the scope for possible future work.

CHAPTER 2

BACKGROUND

IoT is heavily affecting our daily lives in many domains, ranging from tiny wearable devices to large industrial systems. However, ensuring the security of IoT systems is really challenging due to the heterogeneous nature of the IoT environments [57]. IoT has experienced unprecedented attacks that target not only businesses but also have a probability of endangering human lives [58]. Therefore, providing a comprehensive classification of IoT attacks and the available defenses against them is the first step towards securing IoT systems. This chapter presents a comprehensive survey on IoT security attacks and provides a taxonomy to classify them based on different aspects, namely, architecture, location and access level of the attacks, data damage level, objects of vulnerability, and IoT security requirements. This thorough survey provides valuable insights for industry and academic researchers engaged in the design of secure IoT systems. It serves as a valuable resource, aiding in identifying the areas that have been extensively studied and those that require further exploration. By leveraging the findings of this survey, researchers can enhance their understanding of attack vectors, prioritize

their efforts, and develop robust countermeasures to fortify IoT systems against emerging threats. This chapter discusses the potential IoT security attacks based on different aspects namely, architecture, location and access level of the attacks, data damage level, and objects of vulnerability. The taxonomy of IoT security attacks is also highlighted.

2.1 IoT Security Attacks

IoT is defined as interconnected networks where IoT entities (smart objects, devices, people, etc.) interact with each other to exchange information through communication channels [5]. In contrast to IoT infrastructure, the architecture design of conventional security is based on the user view and it is not applicable for communicating between machines [57], [59]. Although conventional and IoT infrastructures have similar security issues, however, different techniques could be applied to handle each security issue. Table 2.1 shows the key differences between IoT security and conventional security in terms of the following characteristics: security design, device capability, resources, node energy, the heterogeneous feature, security guards and environment.

Table 2.1: Key Differences between Conventional and IoT Security

Characteristic	Conventional Security	IoT Security
Security Design	Add-on Security	Built-in Security
Device Capability	Complex	Lightweight
Resources	Rich	Limited
Node Energy	Abundant	Constrained
Heterogeneity	No	Yes
Security Guards	Many	Few
Environment	Closed	Open

IoT design model has been divided into three layers: application layer, transportation layer, and perception layer as shown in Figure 2.1. Each layer is a potential target for several designated attacks. The following sections discuss the potential IoT security attacks based on different aspects.

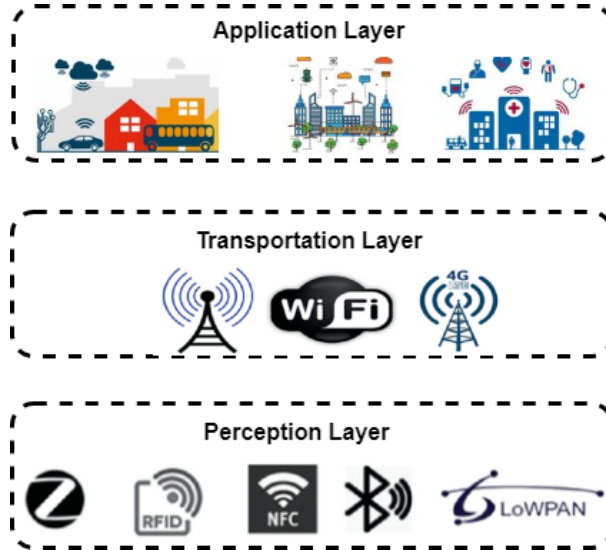


Figure 2.1: IoT Design Model

2.2 Application Layer Attacks

The application layer is a place where specific application business situations Smart City, Smart Factories, Health Care etc. Each business environment might have its own security and privacy challenges since there is no security standard at IoT for the application layer. Furthermore, some security threats at this layer cannot be avoided at other layers such as privacy and protection. Even though this layer may regularly have a complex logical business in different environments with different purposes, it is vulnerable to the following common security threats [60]–[62]:

1. **Data Leak**

A data leak occurs when information intended to remain private or protected is accessed, copied, or distributed without permission. Data leaks can have severe consequences, leading to compromised personal privacy, financial loss, reputational damage, or even legal implications. Typically, data leaks occur due to various factors, such as cyberattacks, system vulnerabilities, insider threats, or human error [30].

2. **Denial-of-Service (DoS) Attack**

DoS attack refers to a malicious action that aims to render a network, computer system, or service inaccessible to its legitimate users. The attacker executes a DoS attack by inundating the targeted system with an overwhelming volume of unauthorized requests or an excessive amount of traffic, depleting its resources and resulting in unresponsiveness or system failure.

This disruption in service may have significant consequences, such as loss of productivity, financial losses, and damage to an organization’s reputation. Various techniques can be employed to carry out a DoS attack, such as flooding the target with excessive network traffic, exploiting system vulnerabilities, or overwhelming it with resource-intensive processes [63], [64]. In this type of attack, application service availability could be targeted to thousands of fake requests in order to block or shut down the service from other legitimate customers. Flooding attack in the cloud is classified as one type of DoS attack that may lead to exhaust the resources and have an effect on the quality of service in the cloud [65], [66].

3. Malicious Code Injection (MCI)

Malicious code injection is a common technique used by attackers to exploit vulnerabilities and compromise the security of a target system. This injection can occur through various means, such as input fields, file uploads, or insecure APIs. Once the malicious code is injected, it can execute arbitrary commands, steal sensitive data, manipulate system behavior, or provide unauthorized access to the attacker. Common examples of malicious code injection include remote code execution and cross-site scripting [65]–[70].

4. Buffer Overflow Attack

A buffer overflow attack refers to a security flaw that arises when a program or process attempts to store an excessive amount of data in a buffer, surpassing its allocated capacity. This overflow can overwrite adjacent memory

locations, creating a potential avenue for executing malicious code or triggering program crashes and unpredictable behavior. Attackers exploit buffer overflow vulnerabilities by deliberately inputting excessive data, often with carefully crafted malicious code, to manipulate the execution of the program or obtain unauthorized entry into the system. Buffer overflow attacks can have severe consequences, allowing attackers to take control of a targeted system, execute arbitrary commands, or launch further attacks [64], [68].

5. **Malware Injection**

Malicious software like Viruses, Ransomware, Worms, Trojans, Spyware, Adware and Rootkits that could be installed in devices and networks and therefore making damage to them or getting/losing the system control. Malware injection is a common tactic employed by cybercriminals to compromise the security and integrity of targeted systems. Malware can infiltrate systems through multiple channels, including exploiting software vulnerabilities or deceiving users into downloading infected files or clicking on malicious links, or compromising legitimate websites to distribute malware to unsuspecting visitors. Once injected, malware can execute a diverse array of malicious actions, such as exfiltrating sensitive information, gaining unauthorized access, causing disruptions to normal system operations, or propagating further malware infections [64], [71].

6. **Phishing**

Phishing attacks typically involve sending deceptive emails, messages, or

websites that closely resemble authentic and trustworthy entities, such as financial institutions, social media platforms, or e-commerce websites. The goal is to deceive recipients into believing that the communication is genuine and prompting them to disclose their confidential information. These phishing attempts often employ psychological manipulation, urgency, or fear tactics to increase the chances of success. Phishing attacks have severe consequences, including identity theft, financial loss, or unauthorized access to personal accounts [64], [65], [67], [71].

7. SQL Injection

SQL injection is a security weakness that arises when a malevolent actor maliciously inserts unauthorized SQL code into input fields or parameters of a web application that interacts with a database. This vulnerability allows the attacker to tamper with the application's SQL statements, thereby potentially obtaining unauthorized entry into the database, extracting sensitive information, or modifying its contents. SQL injection attacks pose a significant threat due to their ability to bypass authentication mechanisms and directly target the underlying database, posing a substantial risk to the accessibility, integrity, and confidentiality of the data [64], [68], [71], [72].

2.3 Transportation Layer Attacks

The transportation layer is responsible for delivering all kinds of communication traffic between the applications and their related outsourced peripherals such as

sensors, actuators, networking devices, etc., which can be connected through either wireless or wired mediums. The attacks on the network layer are different due to the variety of protocols and devices used in different networks. Transportation layer is a potential target to the following attacks [30], [60]–[62], [64], [65], [67], [73]–[76]:

1. Routing Attack

A routing attack is a type of cybersecurity threat that may occur when an attacker interferes with the routing of network traffic to redirect it to unauthorized destinations or intercept sensitive information. This attack targets the routing infrastructure, such as routers or switches, to manipulate the paths that data packets take within a network. By compromising the routing protocol or gaining unauthorized access to routing devices, an attacker can reroute traffic to malicious servers or intercept and inspect the data packets for sensitive information. Routing attacks can have severe consequences, including unauthorized access, data theft, surveillance, or disruption of network connectivity [67], [73], [74], [77]–[80].

2. DoS Attack

The heterogeneous nature of IoT makes the communication layer a potential target to DoS attacks [30], [64]–[67], [70], [71], [74], [78], [79]. Please refer to Section 2.2.2.

3. Data Transit Attack

In this kind of attack, the attackers position themselves between the sender and the recipient, secretly capturing and manipulating the data being transmitted. This can be achieved through various techniques, such as eavesdropping on unencrypted network traffic, impersonating legitimate network devices, or exploiting vulnerabilities in encryption protocols. The goal of a data transit attack is typically to gain unauthorized access to sensitive information including financial records, login credentials, or personal data [67], [81].

4. **Replay Attack**

In this attack, the attacker captures legitimate data or commands exchanged between IoT devices and the network, and then resends them at a later time to gain unauthorized access or control over the targeted devices. This can lead to various security risks, such as unauthorized device access, unauthorized data retrieval, or unauthorized actions performed by IoT devices [64], [78].

5. **Sybil Attack**

In a Sybil attack, the attacker masquerades as multiple distinct entities, often by creating numerous bogus accounts or nodes, and uses them to control or influence the network's behavior. This attack can be particularly problematic in decentralized networks, such as peer-to-peer networks or blockchain systems, where trust and consensus are critical. Sybil attacks can lead to various consequences, including the ability to manipulate voting or consen-

sus mechanisms, launch spam or DoS attacks, or gain undue influence over network activities [64]–[66], [72]–[74], [78]–[80], [82]–[84].

6. Sinkhole Attack

In this attack, the attacker manipulates the routing infrastructure or Domain Name System (DNS) to redirect legitimate traffic to the sinkhole, where it can be intercepted, monitored, or tampered with. Sinkhole attacks can be used for various malicious purposes, such as capturing sensitive information, launching further attacks, or disrupting network communications. By redirecting traffic to the sinkhole, the attacker can intercept data, perform MITM attacks, or gain unauthorized access to systems or devices [62], [64]–[66], [74], [77], [78], [80], [83], [84].

7. Session Hijacking

The attack typically occurs by exploiting vulnerabilities in the session management process or by capturing the session identifier, such as a session cookie or token. Once the attacker gains control of the session, they can impersonate the legitimate user, access sensitive information, perform unauthorized actions, or manipulate the user’s session data. This attack is particularly dangerous because it allows the attacker to bypass authentication mechanisms and directly exploit the user’s active session [85], [86].

8. MITM Attack

MITM in the context of the IoT refers to a cybersecurity threat where an at-

tacker intercepts and manipulates the communication between IoT devices or between IoT devices and the network. In a MITM attack, the assailant strategically positions themselves between the communicating parties, surreptitiously capturing and altering the exchanged data. This malicious tactic enables the attacker to covertly eavesdrop on sensitive information, tamper with data integrity, inject malicious commands or even impersonate legitimate devices. MITM attacks pose a significant threat to the confidentiality, integrity, and availability of IoT systems, leading to unauthorized access, data theft, or unauthorized control over devices.

9. **Advanced Persistent Threat (APT)**

APT attacks are characterized by their persistence, as they are specifically crafted to avoid detection for prolonged durations, sometimes spanning weeks, months, or even years. APT attackers employ advanced techniques, such as social engineering, exploiting zero-day vulnerabilities, targeted spear-phishing, and developing custom malware, to breach network defenses and establish a foothold within the targeted environment. Once inside, they aim to maintain persistence, gather intelligence, and progressively escalate their privileges to achieve their objectives, which may include data theft, espionage, or sabotage [67], [71].

10. **Sniffing Attack**

In a sniffing attack, the attacker uses specialized software or hardware tools to monitor and capture data packets that are transmitted over a network.

By analyzing the captured packets, the attacker can extract valuable information, such as usernames, passwords, credit card details, or other confidential data. Sniffing attacks are particularly effective when the network traffic is not encrypted or when the attacker has access to a network segment where the traffic flows [64], [84].

11. Traffic Analysis

Traffic analysis focuses on many factors such as packet size, timing, source and destination addresses, and communication protocols. By analyzing this information, an attacker or analyst can derive valuable intelligence, such as identifying communication patterns, determining the type of applications being used, inferring user behavior, or even uncovering sensitive information. Traffic analysis can be used for legitimate purposes, such as network optimization or monitoring for security threats, but it can also be leveraged by attackers to gather intelligence about a target, identify vulnerabilities, or launch more sophisticated attacks [64], [74], [78], [80].

2.4 Perception Layer Attacks

The perception layer of IoT basically involves gathering and processing data over WSNs, Global Positioning System (GPS) and Radio-Frequency Identification (RFID) and RFID Sensor Network (RSN). The perception layer consists of sensors and actuators that aim to either query the location or measure, for example, temperature, acceleration, humidity, etc. Major security attacks targeting

the perception layer are as follows [60], [61]:

1. **Physical Attack**

Physical attacks can take various forms, such as theft, vandalism, tampering with devices or cables, or gaining unauthorized physical access to sensitive areas. These attacks can have severe consequences, including data theft, unauthorized access, system disruption, or even the destruction of critical infrastructure [69], [74], [79], [80], [82], [84].

2. **Impersonation**

Impersonation attacks can occur through various means, including social engineering techniques, phishing emails, or the creation of fake accounts or websites that mimic trusted entities. The goal of impersonation is often to deceive individuals with the intention of extracting confidential information like personal data and login credentials, or to illicitly obtain unauthorized access to systems and valuable resources. Impersonation attacks may have significant consequences, including identity theft, financial loss, or unauthorized access to sensitive information [87].

3. **Eavesdropping**

Eavesdropping involves surreptitiously capturing or monitoring network traffic or communication signals to gain unauthorized access to information. Eavesdropping attacks can occur in various scenarios, including unsecured Wi-Fi networks, unencrypted communication channels, or compromised net-

work infrastructure. The attacker aims to gather valuable information, such as usernames, passwords, financial data, or confidential business information [66], [70].

4. **DoS Attack**

DoS attack refers to a malicious action that aims to render a network, computer system, or service inaccessible to its legitimate users. Due to the heterogeneous nature of the IoT environments with limited resources, the perception layer may be also a potential aim for DoS attacks [30], [73], [88].

Please refer to Section 2.2.2

5. **Jamming Adversaries**

Jamming refers to the deliberate transmission of interference signals or noise on the same frequency bands used by wireless devices, rendering the communication signals ineffective or unusable. Jamming attacks can be carried out using various techniques, such as transmitting high-power signals, employing frequency-hopping techniques, or utilizing wideband interference. The goal of jamming adversaries is to disrupt wireless communication systems, prevent legitimate users from accessing networks or services, or interfere with critical infrastructure [64], [65], [89], [90]. This type of threat may lead to downtime in the IoT environment due to improper connection to an authorized node [66], [73], [78], [83].

6. **Deprivation Attack**

In a deprivation attack, an adversary overwhelms a system’s resources, such as processing power, memory, network bandwidth, or storage capacity, by continuously sending a high volume of legitimate or malicious requests. As a result, the targeted system becomes overwhelmed, leading to degraded performance or complete unavailability. Deprivation attacks can have significant consequences, including disrupting critical services, causing financial losses, or impacting the reputation of targeted organizations [64], [67], [79], [80], [82], [91]–[93].

7. Node Capturing

Node capturing, in the context of cybersecurity, refers to a type of attack where an adversary gains control over a node within a network, typically in a decentralized or peer-to-peer system. The captured node can be a legitimate participant in the network or a malicious actor that impersonates a legitimate node. Once a node is captured, the attacker can manipulate or disrupt network activities, modify or forge data, launch further attacks, or exploit the compromised node for unauthorized purposes. Node capturing attacks can be particularly damaging in distributed systems, such as blockchain networks or peer-to-peer file-sharing systems, as they can undermine trust, compromise consensus mechanisms, or compromise the integrity of the network [67], [68], [70], [74], [80], [94].

8. **Malicious Code Injection (MCI)** An attacker may exploit a vulnerable node for example, with maliciously crafted inputs in the memory of the

node to push it to an action that it is not intended to do [67], [72], [74], [78], [88]. The ultimate goal of this attack is to take control of the whole IoT system [80], [80], [84], [95]. Please refer to Section 2.3.3.

9. **Side-Channel Attack** Unlike traditional attacks that target vulnerabilities in software or hardware, side-channel attacks focus on analyzing the physical characteristics or observable behaviors exhibited by a system, encompassing factors like power consumption, electromagnetic emissions, timing measurements, or even sound. Through the examination of these side-channel signals, an adversary can infer confidential information, such as encryption keys or passwords, without directly attacking the cryptographic algorithms themselves. Side-channel attacks can be particularly challenging to defend against since they often exploit inherent physical properties or implementation flaws rather than traditional software vulnerabilities [67], [78], [82], [84].

Table 2.2 summarizes the IoT security attacks with the corresponding affected layers from the perspective of the IoT design model shown in Figure 2.1.

Table 2.2: Summary of Main Existing Attacks in IoT Model

Layer	Attack	Reference
App.	Data Leak	[30]
	Denial-of-Service	[65], [66]
	Malicious Code Injection	[65]–[70]
	Buffer Overflow	[64], [68]
	Malware Injection	[64]
	Phishing	[64], [65]
	SQL Injection	[64], [68], [72]
Trans.	Denial-of-Service	[30], [64]–[66], [66], [70], [81]
	Phishing	[67]
	Routing Attack	[67], [73], [74], [78]–[80]
	Data Transit	[67], [81]
	Replay Attack	[64], [78]
	Sybil Attack	[64], [65], [73], [74], [78], [80], [83]
	Sinkhole Attack	[62], [64], [65], [74], [78], [80], [83], [84]
	Session Hijacking	[85], [86]
	Man-in-the-Middle	[64], [65], [70], [72], [74], [78], [80], [83]
	Advanced Persistent Threat	[67]
	Sniffing Attack	[64], [84]
	Traffic Analysis	[64], [74], [78], [80]
	Jamming Adversaries	[83]
Perc.	Denial-of-Service	[30], [81], [88]
	Malicious Code Injection	[67], [72], [74], [78], [88]
	Sybil Attack	[72]
	Physical Attack	[69], [74], [79], [80], [82], [84]
	Impersonation	[87]
	Eavesdropping	[66], [70]
	Jamming Adversaries	[64]–[66], [73], [78], [83]
	Deprivation Attack	[64], [67], [79], [80], [82], [93]
	Node Capturing	[67], [68], [70], [74], [80]
	Side-Channel Attack	[67], [78], [82], [84]

2.5 Attacks Based on Access Level

1. Passive Attacks

Passive attacks, in the context of cybersecurity, are a type of attack that focuses on unauthorized access or information gathering without altering or disrupting the targeted system or communication. Passive attacks aim

to passively eavesdrop, intercept, or monitor network traffic or information without leaving a trace. These attacks can include activities such as network sniffing, packet capture, or traffic analysis to gather sensitive data or gain intelligence. Passive attacks can be challenging to detect since they do not directly impact the targeted system's functionality [73], [96].

2. Active Attacks

Active attacks, in the context of cybersecurity, refer to a type of attack where an adversary actively interacts with or manipulates the targeted system or communication to disrupt, modify, or gain unauthorized access. Unlike passive attacks that focus on information gathering without altering the system, active attacks involve actions that directly impact the integrity, availability, or confidentiality of the targeted system or data. Examples of active attacks include malware infections, unauthorized system access, data alteration, or DoS attacks. Active attacks can range from relatively simple exploits to highly sophisticated techniques employed by skilled adversaries [73], [96], [97].

2.6 Attacks Based on Geographic Location

1. Insider Attacks

Insider attacks are a type of cybersecurity threat that originate from within an organization or system, involving individuals who have authorized access to sensitive resources. These insiders could be employees, contractors, or

business partners who abuse their privileges or exploit their trusted position to carry out malicious activities. Insider attacks can take various forms, such as data theft, sabotage, unauthorized access, or the introduction of malware or backdoors. Insider attacks can be particularly damaging as they have legitimate access to critical systems and sensitive information, making detection more challenging [73].

2. Outsider Attacks

Outsider attacks, in the context of cybersecurity, refer to threats that originate from external sources or unauthorized individuals without legitimate authorization seek to gain access to an organization's systems or resources. These attackers are typically external actors, such as hackers, cybercriminals, or state-sponsored groups, who make unauthorized attempts to gain access, exploit vulnerabilities, or compromise systems for various malicious purposes. Outsider attacks can take different forms, including phishing, malware infections, network intrusion, DoS attacks, or exploitation of software vulnerabilities [73].

2.7 Attacks Based on Data Damage Level

1. Interception

Unauthorized individuals may gain access to confidential data or the environment. Interception could be conducted against any stage of the data life cycle (i.e data at rest, data in use, data in transit). Examples of interception

attacks are sniffing attacks, traffic analysis, eavesdropping, data leak, and side-channel attack [98], [99].

2. Interruption

Interruption attacks may cause a network service or an asset of the system to become unavailable for legitimate use. Interruption could be conducted against data in transit. Examples of interruption attacks include DoS attack, sinkhole attack, deprivation attack, jamming adversaries, and physical attack [98], [99].

3. Fabrication

Fabrication attacks may generate illegitimate data, communications, processes, or other fake items within IoT systems/networks. Fabrication attacks could be conducted against any type of data life cycle stages. Examples of fabrication attacks are MITM attacks, routing attacks, impersonations, and SQL injection [73], [98], [99].

4. Modification

In modification attacks, an unauthorized party not only gains access to the data or an asset but also tampers with it. Modification attacks could be conducted against any stage of the data life cycle. Examples of modification attacks are data manipulation attacks, MITM attacks, and routing attack [73], [98], [99].

2.8 Attacks Based on Objects of Vulnerability

1. Hardware (Physical) Attacks

An attacker could stay physically close to the IoT devices/network to launch the physical attack. The most common types of physical attacks are tampering, jamming adversaries, side channel attack, and deprivation attack [78], [100].

2. Software Attacks

In this type of attack, software vulnerabilities could be exploited to launch the attack. Examples of software attacks are viruses, worms, adware, spyware, Trojan horses and malware injection [101].

3. Network Attacks

Network attacks are carried out to cause damage to IoT network systems. Network attacks may be launched even if they are not close to IoT networks. The common types of network attacks are traffic analysis attacks, routing attacks, MITM attacks, sinkhole attacks, Sybil attacks, DoS attacks, and replay attacks [78], [100].

4. Data Attacks

Numerous security attacks aim to expose sensitive stored or transmitted data and break the security authentication mechanisms in IoT environments. Examples of data attacks are data inconsistency, unauthorized access, data leak and buffer overflow [78].

2.9 IoT Security Attacks Taxonomy

IoT security attacks can be classified based on many aspects as shown in Figure 2.2. Different studies classified IoT attacks based on the three-layered reference architecture [30], [60], [62], [66]. Other studies classified the IoT attacks based on different architectural reference models (i.e., four/five layers) [64], [65], [67], [69], [75], [80]. Another IoT attacks taxonomy was based on a novel building-blocked reference model which composes of four main phases (cloud, network, edge computing and perception) [84]. In addition, various studies classified IoT attacks and threats in specific environments and technologies such as WSN and RFID [30], [92].

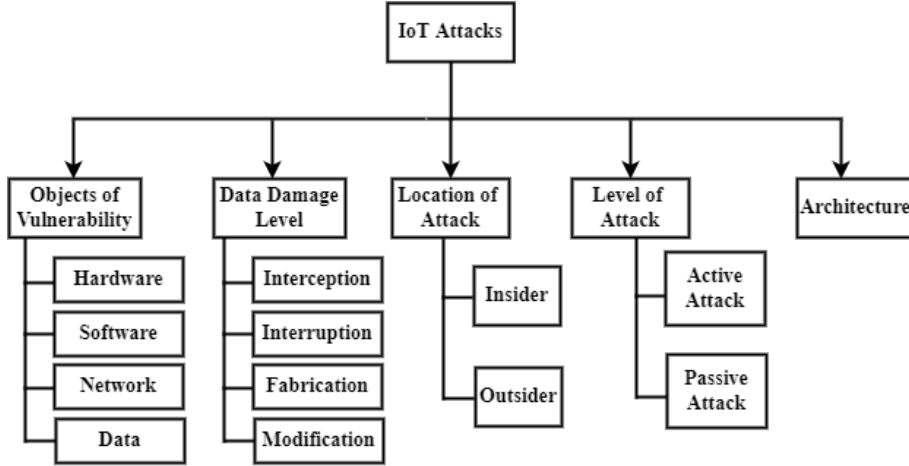


Figure 2.2: IoT Security Attacks

On the other hand, several studies categorized IoT attacks based on different aspects such as access level of the attacks (Active or Passive) [73], [102], location of the attacks (Insider or Outsider) [73], data damage level (Interception, Interruption, Fabrication, Modification) [73], objects of vulnerability (i.e. Hardware,

Software, Network, Data) [73], [78]. Another factor to classify IoT security attacks is based on their impact on the IoT security requirements [103]. Table 2.3 summarizes the existing taxonomies for IoT security attacks.

Table 2.3: Summary of Existing IoT Attacks

Attack Based on	Papers
Three-layered Reference Architecture	[30], [60], [62], [66]
Four/Five-layered Reference Architecture	[64], [65], [67], [69], [75], [80], [84]
Architectural Layers for WSN & RFID	[30], [92]
Level of Attack	[73], [96]
Location of Attack	[73]
Data Damage Level	[73]
Objects Of Vulnerability	[73], [78]

CHAPTER 3

INTERNET-OF-THINGS DATA

SECURITY

Data security is of utmost significance for organizations. Safeguarding sensitive information has become increasingly crucial in today's digital landscape. Protecting data ensures the preservation of an organization's reputation, customer trust, and overall operational integrity. By implementing robust security measures, organizations can mitigate the risks of data breaches, unauthorized access, and malicious activities. The occurrence of data breaches can lead to significant financial ramifications, legal repercussions, and substantial harm to the reputation and image of the organization. Moreover, compliance with data protection regulations is essential to avoid penalties and maintain legal compliance. Prioritizing data security enables organizations to maintain a competitive edge, foster customer loyalty, and establish a strong foundation for sustainable growth. This chapter elaborates on the need for data security and emphasizes its importance for

organizations. This chapter also discusses the stages of data security with their challenges and countermeasures especially in IoT environments. In addition, this chapter highlights the taxonomy of IoT countermeasures.

3.1 Overview

Over the past few years, the emergence of the IoT has brought about a significant shift in human-machine interaction [2]. This paradigm shift has revolutionized various aspects of our lives by enabling physical objects to communicate and collaborate in order to accomplish specific tasks [3]. The IoT utilizes advanced technologies such as Internet protocols, WSNs, and embedded devices to make objects more intelligent [4], [5]. As a result, it has become a popular trend with far-reaching implications for business operations in the foreseeable future [5].

However, the rapid advancement of IoT also presents considerable challenges in terms of data security for designing and implementing IoT applications. Since IoT heavily relies on internet infrastructure, it becomes susceptible to both traditional and cybersecurity threats. Cyberattacks targeting IoT not only compromise the functionality of IoT systems but also pose risks to human safety and privacy. While security countermeasures have been developed since the early stage of the internet era, they cannot be directly applied to IoT due to the significant differences in computation, power, and storage capabilities [7].

IoT has witnessed unprecedented attacks which not only affect businesses but also pose potential risks to human lives. Therefore, it is crucial to develop a

comprehensive understanding of the threats and available countermeasures. This chapter conducts an extensive survey on IoT challenges and countermeasures, and proposes a taxonomy to categorize them based on the three data states (i.e. data at rest, data in use, and data in transit). This chapter also discusses future directions for research in this area to develop more effective and efficient security solutions.

3.2 Data States Security Challenges

According to the 2020 conceptual framework [25], it is clear from the IoT architecture expressed in Equation (3.1) that data is one of the four vital components in any IoT system [104].

$$IoT = Services + Data + Networks + Sensors \quad (3.1)$$

In general, data can be classified into three different states, namely, data at rest, data in use, and data in transit [37], [105], [106] as shown in Figure 3.1. Data at rest refers to inactive data that is physically stored within storage units such as data warehouses, databases, spreadsheets, tapes, mobile devices, embedded Multi-Media Card (eMMC) storage devices, etc., [106]–[111]. Data in use refers to actively manipulated data that is stored in a temporary digital state, often residing in volatile memory locations such as RAM, CPU registers, or caches [106], [108], [112]. Data in transit referring to data that is traversing any type of private or public communication medium such as internet [37], [107], [108], [110].

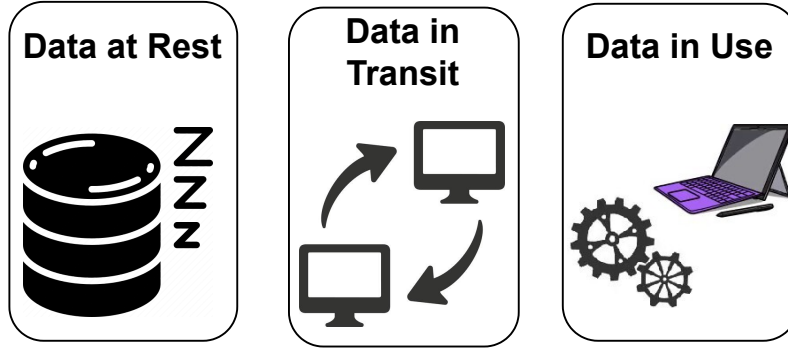


Figure 3.1: The Three Data States

Data serves as a fundamental component within IoT systems, and the primary objective of any computing system is to effectively handle this data [26], [27]. In an IoT environment, data creation, computation, and management often occur beyond the confines of traditional data centers. This presents a significant challenge as it places the burden on IoT devices to perform resource-intensive computational tasks, potentially resulting in substantial delays in service response times [27]–[29]. Furthermore, in order to efficiently manage data, certain IoT devices may store copies of collected data within their local internal storage. However, this practice introduces a new set of security vulnerabilities, including the risk of data leakage and data theft [45], [113]. The presence of local data copies can make IoT devices an attractive target for malicious actors seeking to exploit these vulnerabilities.

IoT comprises a network of interconnected devices that collaborate to generate, store, and process data across multiple physical locations. These devices are not confined to a local domain where we have full control. Consequently, this necessitates the adoption of various communication protocols and intermediate gateways, which can potentially impede the implementation of certain security

measures [45], [114], [115]. Another pressing issue revolves around the challenge of effectively managing and identifying vulnerabilities in IoT devices, particularly in light of the constant emergence of critical threats on a daily basis [29], [116]. Controlling and detecting these vulnerabilities poses significant difficulties, further exacerbating concerns regarding the overall security of IoT systems.

Data at rest presents numerous challenges that warrant attention. One prominent issue lies in the shared utilization of storage among multiple users. This arrangement introduces significant risks, including data leakage, unauthorized data modification, and data theft, which can compromise the security of the stored information [78], [117]–[122]. Additionally, the physical location of data adds to the complexity. Data may reside on diverse media or be distributed across various physical sites. This dispersion raises concerns regarding the ability to exercise control over data storage and comply with different data security regulations. Failing to address these risks adequately can undermine the security and integrity of data at rest [110], [123]. Moreover, there exist numerous challenges concerning data in use. Data in use typically pertains to data undergoing processing, thereby increasing the potential for risks throughout various data operations, such as creation and deletion. Data privacy poses a significant obstacle when it comes to data in use, as the data owner must guarantee that the information remains confidential and is not shared with any external parties [78], [124], [125]. An additional peril that arises in relation to data in use is the issue of data remanence. This predicament arises when deleted data can be recovered due to inadequate deletion methods. Insufficiently eradicating data poses a substantial risk to the integrity and confi-

dentiality of sensitive information [123]. Furthermore, data in transit is even more vulnerable compared to data in use and data at rest. [19], [20], [32], [126]. Extensive literature has addressed numerous challenges associated with data in transit, including the diverse range of channels and communication methods utilized [45]–[47], [122], as well as the inherent difficulty in making decisions regarding which data should be protected and which data can be left unprotected [110], [127]. Table 3.1 provides a comprehensive overview of the typical challenges encountered in ensuring data security in IoT. This table serves as a summary of the common obstacles faced when safeguarding data in IoT environments.

Table 3.1: Challenges of Data Security in IoT

Data State	Challenges
Data at Rest	- Shared utilization of storage [78], [117]–[122]. - Data location/Different physical locations [110], [123]. - Inability to control data storage [110], [123]. - Compliance with data security regulations [110], [123]. - Managing/identifying vulnerabilities [29], [116].
Data in Use	- Data Privacy [78], [124], [125]. - Data Remanence Problem [123].
Data in Transit	- Channels/communication diversity [45]–[47], [122]. - Managing/identifying vulnerabilities [29], [116].

3.3 Data at Rest Countermeasures

Various security countermeasures have been proposed to safeguard the security of data at rest. A new efficient and scalable framework based on asymmetric encryption technique to protect data involved in the cloud was proposed in [128]. The proposed encryption technique in this framework is based on factorizing large prime numbers which is an open problem in mathematics.

Encryption techniques could be applied for securing the sensitive data stored in any database or physical storage to add an additional security layer to ensure the security of that sensitive data when the access control techniques (passwords and username) fail. Based on this vision, a novel method was proposed in [129] to safeguard the security of the data at rest in cloud systems. An approach based on Fully Homomorphic Encryption (FHE) was introduced to protect data at rest in cloud banking systems supporting many famous languages like Arabic, English, Chinese, etc. [130]. Another study aimed to reduce the need of extra storage space was introduced in [131]. The study suggested filtering the data at rest and then applying mitigation techniques such as biometric or hashing algorithms to the most important/sensitive data.

A comprehensive study on security countermeasures, guidelines, and threats of data at rest was introduced in [103]. The study suggested many security measures to protect data at rest such as Deduplication, Encryption, Blockchain, Access control, Anonymization, Recovery policy, Distributed data depositories, Physical security and secure data migration. Moreover, the study proposed a framework that

illustrates the relationship between the suggested security measures, attacks, and guidelines. Another comprehensive study identified IoT attacks targeting data at rest that may be resided either in IoT objects or in the cloud [84]. The study also suggested numerous countermeasures such as encryption and key management techniques, backup and retention strategies, digital signature, modern hashing algorithms, homomorphic encryption, protection for live migration of VMs, dynamic credentials, firewall, trusted cloud computing platform, brute-force scanners and lockout techniques. Recently, novel approaches were proposed in [132], [133] to protect data at rest using hardware-based security and passwordless based cryptographic method, respectively.

A new big data security technique was proposed to protect the confidential data in NoSQL database stores such as Mongo databases by integrating encryption with data fragmentation using natural language processing detection techniques [134]. Another database security approach was proposed in [135] to protect data stored in a relational database using Hill encryption technique and XOR based operation.

A novel and dynamic scalable model was introduced in [105] to safeguard the security of data at rest in the fog environment and reduce the computation overload of the fog node as well. A simple security model to achieve the security in IoT applications was proposed in [136]. The study categorized the data involved in IoT into public and private data. Furthermore, the private data was classified into stored data and data in transit. The study suggested some mitigation techniques such as encryption and data attestation, authentication and authorization

to protect both the stored data and data in motion.

A new scalable security approach to protect the data at rest was proposed in [37]. The suggested approach gives the user the opportunity to choose the appropriate level of security based on the classification of data, enabling the implementation of tailored protective measures commensurate with the sensitivity and importance of the information. [137].

Information hidden techniques can also be utilized to enhance the security of data at rest. The authors in [138] proposed a novel technique for protecting the data at rest, utilizing customizable cloaking sequences that empower users to selectively safeguard their information. Table 3.2 summarizes violated security goals, attacks, and countermeasures of the existing studies for data at rest.

3.4 Data in Use Countermeasures

Few studies have been suggested for enhancing the security of data in use. A general-purpose computation approach using Binary Decision Diagrams (BDD) was proposed to protect the data in use. The proposed approach depends on homomorphic operations utilizing the features of BDD in order to transform the Boolean functions [139]. Another study introduced an approach to assess the processes of data leakage protection model for data in use [140]. Homomorphic encryption could also be used to protect data in use [103].

The authors in [141] introduced an extensive security architecture known as FlexSEE, which is specifically designed for confidential computing in contempo-

Table 3.2: Summary of Existing Studies for Data at Rest

Ref	V.G.	Attacks	Countermeasure	
			Solution	Technique
[105]	-	-	P	Encryption
[37]	All	Unauthorized Access, Data Theft	P	Encryption
[128]	All	Unauthorized Access, Timing, Brute Force	P	Encryption
[129]	-	-	P	Encryption
[130]	-	-	P	Encryption
[131]	All	Unauthorized Access	P D/P D/P	Data Filtering Biometric Hashing
[103]	All	Misuse of Data Remnants, Homogeneity, Data Manipulation, Linkage, Insider, Side-channel, DoS, Identification, Unauthorized Access, Hash Collision	P P P P P/M M P P	Deduplication Access Control Anonymization Encryption Blockchain Recovery Policy Distributed Data Depositories Physical Security
[84]	All	Account Hijacking, Data Leak, Data Scavenging, Data Manipulation, Hash Collision, Brute-Force, DoS	P M D/P D/P P P D P/M	Encryption Backup Digital Signature Hashing Dynamic Credentials Firewall Brute-Force Scanners Lockout Techniques
[132]	-	-	P/M	Hardware-Based Security
[133]	All	Unauthorized Access	P	Passwordless Cryptography
[134]	C & I	Data Manipulation	P P	Encryption Data Fragmentation
[135]	-	-	P	Encryption
[136]	-	-	P P P	Encryption Authentication Authorization
[138]	C	Unauthorized Access	P	Data Cloaking

C: Confidentiality I: Integrity D: Detection P: Prevention M: Mitigation V.G.: Violated Goals

rary cloud environments. Unlike traditional approaches, FlexSEE doesn't rely on the trustworthiness of system software running on the compute server. It ensures

that user data is only accessible in a non-privileged mode by a designated program authorized by the data owner and running on designated hardware. This mechanism effectively safeguards the data from untrusted hardware, hypervisors, operating systems, or other applications belonging to different users on the compute server. By implementing FlexSEE, the confidentiality of sensitive information is significantly enhanced.

Another study [142] concentrated on NoSQL document databases and presented a novel protection model designed to safeguard sensitive data during its usage within the database. The proposed approach seamlessly and efficiently encrypts sensitive data prior to its storage in the database. The results obtained from this study demonstrate that the proposed model introduces an additional layer of security, which can be regarded as a standard solution for mitigating potential attacks on an active database instance. By implementing this model, the confidentiality of sensitive data is significantly enhanced, providing a reliable defense against potential threats. Table 3.3 summarizes the violated security goals, attacks and countermeasures of the existing studies for data in use.

Table 3.3: Summary of Existing Studies for Data in Use

Ref	V.G.	Attacks	Countermeasure	
			Solution	Technique
[139]	All	Brute-force, Inverse Circuit, Boolean Satisfiability	D/M	Binary Decision Diagrams
[140]	All	Insider	D/P	Data Leakage Protection
[141]	All	Unauthorized Access	P	Encryption
[142]	All	Unauthorized Access	P	Encryption

D: Detection P: Prevention M: Mitigation V.G.: Violated Goals

3.5 Data in Transit Countermeasures

Many studies have been proposed to introduce different security solutions to protect data in transit. A validation algorithm based on hash function was proposed to guarantee the security of data during transmission and thwart any unauthorized interception attempts or any objects injection in cloud computing environments [43]. The proposed validation algorithm could also be used to mitigate MITM, which is one of the serious authentication threats in data transmission [143]–[145]. A new Data Leakage Prevention (DLP) architecture for data in motion was proposed by [146]. The proposed approach utilizes machine learning techniques to classify the data into non-confidential and confidential, and then applies encryption techniques to protect the confidential data, and subsequently ensuring efficient hardware resources.

A heterogeneous ring signcryption approach was introduced to establish a robust layer of protection for the data exchange between servers and sensors. The proposed approach has many special features which make it suitable to protect the data transmission in the IoT [147], [148]. A data sharing protocol named certificateless signcryption was proposed to secure data transmission with minimal overhead [149]. The strength of the proposed protocol’s security stems from the complexity of bilinear-DHI and Diffie-Hellman inversion. A novel approach for enhancing the security of data in transit for Cloud-MANET-IoT framework was proposed to build a secure communication system between IoT nodes [44].

A new approach employing tunable two channel communication was intro-

duced to mitigate many threats such as data injection, MITM attacks, jamming attack, and eavesdropping. The proposed approach applies different data splitting methods using OMNeT++ simulator to attain the targeted level of data security, which is good enough for many military and commercial organizations. The proposed approach overcomes the overhead required by modern cryptography techniques [150]. A new technique was proposed in [151] to enhance the security of data at rest and data in transit using segmentation, shuffling, and multikey encryption. In addition, another approach was introduced in [152] to provide a service-oriented library, to enhance the security of data transmission as well as data at rest, which contributes to increasing the confidentiality of devices on the hardware level. In [153], the authors proposed a new scheme to secure data transmission for big data and industrial IoT. The proposed scheme is based on chaotic secure communication which is an effective way to address and resolve the data transmission security issues within industrial IoT environments.

On the other hand, numerous lightweight approaches have been suggested to enhance communication security in IoT environments. Authors in [48] introduced a Secure Data Transmission Scheme (SDTS) specifically designed for WSNs. This scheme involves a hierarchical structure comprising a base station, cluster heads, and members. The SDTS employs ECC for encrypting and decrypting data transmitted by IoT sensors. It is worth noting that the SDTS offers resistance against replay and brute force attacks. However, a vulnerability exists in the scheme due to the interception of ephemeral public keys, which can be exploited by attackers for MITM attack [154]. Furthermore, the scheme falls short in achieving integrity

and authentication between cluster heads and their respective members [154]. An authentication scheme [49] was introduced for IoT systems based on WSNs, employing ECC and hash functions. The network model encompasses a gateway, users, and sensors, where both users and sensors are required to authenticate themselves to the gateway before initiating data transmission. However, it is crucial to note that this scheme is susceptible to different threats, including brute force and replay attacks. Furthermore, the scheme exhibits high computation and communication costs, making it inefficient for deployment in IoT environments [48].

Authors in [51] presented an authentication scheme that relies on ECC and HTTP cookies to maintain the confidentiality of data transmitted between embedded devices and cloud servers. However, the proposed scheme fell short in achieving both mutual authentication and integrity [50]. Additionally, it suffers from high computation and communication costs [48]. Another authentication scheme [50] was proposed for cloud servers and IoT environments. The proposed scheme is based on ECC and the purpose of its design was to address and overcome the inherent security vulnerabilities and weaknesses in [51]. The primary limitation of this scheme is that it does not ensure data integrity. An authentication scheme [50] was introduced specifically for securing communication between cloud servers and IoT environments. The system parameters are established and managed by the cloud server, and throughout the registration procedure, each device is required to provide its unique ID and password. Following registration, the cloud server generates and transmits a cookie to the embedded device. This authentication scheme is based on ECC and aims to address the security vulnerabilities identified

in [51]. However, the scheme has notable limitations, including the failure to ensure integrity and the presence of high communication and computation costs [48]. These limitations necessitate further enhancements to improve the scheme's effectiveness in guaranteeing data integrity and to optimize its communication and computation efficiency.

In addition, a new authentication scheme [52] was introduced to safeguard data transmission across various devices in IoT environments. The proposed scheme follows a trust-minimized approach and incorporates Physical Unclonable Functions (PUFs) to streamline the authentication process and mitigate cloning attacks. However, it is imperative to acknowledge and emphasize that the computational and communication costs associated with the proposed scheme are substantial, thereby hindering its efficiency in IoT environments [155]. Consequently, there is a need for further enhancements to optimize the scheme's performance and make it more appropriate for IoT deployments. In [53], researchers introduced a lightweight approach for IoT which is based on Pseudonym Based Cryptography, Identity Based Encryption, and ECC. This approach enables efficient message communication and authentication between IoT devices and fog nodes. The scheme successfully achieves authentication and addresses the challenge of intractability among IoT nodes. However, the computational cost of the scheme is relatively high due to the utilization of pairing operations, which are known to be resource-intensive processes [52], [53]. A novel key management scheme [156] was proposed with the aim of preserving data confidentiality during transmission in IoT-enabled WSNs. This scheme addresses a critical security vulnerability found

in [157], which lacks session key agreement and mutual authentication. However, it should be noted that the process of identity authentication requires a substantial amount of energy expenditure [158]. Nonetheless, the scheme offers a potential solution to overcome this limitation and enhance the overall security of IoT-enabled WSNs. Authors in [159] introduced a method to enhance the security of data transmission among a server, a group of sensors, and a third party. The primary objective of this technique is to reduce energy usage and enhance the efficiency of sensors. However, the proposed technique fell short in achieving mutual authentication. Furthermore, it exhibits a substantial communication overhead, making it inefficient for IoT applications based on WSNs. Therefore, there is a need for further improvements to address these limitations and optimize the technique for WSN-based IoT scenarios.

Moreover, a novel approach is presented in [160] to establish a model for data transmission and sinkhole detection. This model incorporates watermarking and homomorphic encryption techniques. The watermarking technique is employed to ensure data authentication, while homomorphic encryption is utilized to enable an efficient means of detecting and preventing sinkhole attacks. However, as indicated in [161], the effectiveness of the model in detecting sinkhole attacks may be limited in certain scenarios. Therefore, further research and improvements are warranted to enhance the model's performance and adaptability in diverse sinkhole attack scenarios. The authors in [126] introduced an innovative Innovative Data-in-Transit Defender (DiTD), which aims to guarantee the security of data during transmission. DiTD offers a unique security framework that harnesses the

capabilities of advanced cloud computing, this protocol stands out by its distinctive approach. It is meticulously crafted to maximize the utilization of symmetric and asymmetric cryptography, hash functions, and key exchange mechanisms. Through these optimizations, the protocol significantly enhances the efficiency and effectiveness of these security components, ensuring a robust and reliable security foundation. Furthermore, Blockchain technology [162] and quantum cryptography protocols [163] are also utilized to provide a safe data transmission for IoT devices. A novel encryption approach based on quantum walk model was proposed to secure data transmission in 5G-IoT environments [163]. Table 3.4 summarizes violating security goals, attacks, and countermeasures of the existing studies for data in transit.

3.6 Countermeasures Taxonomy

Based on the comprehensive survey presented in this chapter, the countermeasures for IoT can be categorized into a taxonomy illustrated in Figure 3.2. This taxonomy organizes data security measures according to the three states of data, along with their corresponding countermeasures. The first state to consider is data at rest, which refers to data residing in various locations such as the data source, fog, or cloud. A range of countermeasures could be applied to enhance the security of data at rest including access control [103], [136], blockchain, deduplication, recovery policy [103], data cloaking [138] and hardware-based security solutions [132]. In addition, encryption and key management techniques play a

Table 3.4: Summary of Existing Studies for Data in Transit

Ref	V.G.	Attacks	Countermeasure	
			Solution	Technique
[146]	C	Data Leakage, Data Theft	P P	Machine Learning Encryption
[147]	C, I	Chosen Ciphertext	P	Encryption
[149]	-	-	P	Encryption
[152]	-	-	P/M	SEcube Platform
[153]	C	Brute Force	P/M	Chaotic Secure Communication
[48]	All	Replay, Sinkhole, Brute Force	P	Encryption
[52]	C, I	Cloning, Physical, Side-Channel	D/P P	Trust Model Encryption
[53]	C, I	Impersonation, Replay, MITM, Forgery	P	Encryption
[156]	All	Impersonation, DoS, Replay	P	Encryption
			D/P	Hashing
[159]	I	Replay	P	Encryption
[160]	A	Sinkhole	P	Encryption
			P	Watermarking
[126]	All	Data Disclosure, Session Hijacking, MITM, Replay, Compromised-key	P	Encryption
[163]	All	Unauthorized Access, Brute Force	D/P	Quantum Walk Algorithm

C: Confidentiality I: Integrity A:Availability D: Detection P: Prevention M: Mitigation V.G.: Violated Goals

major role to protect data at rest. They can be further classified into asymmetric based encryption framework [128], fully homomorphic encryption [130], biometric and hashing algorithms [131], passwordless based cryptography [133], digital signatures [84] and other encryption-based models and protocols [37], [105], [129], [134], [135]. The second state is the data in use where different security solutions could be used to protect data in use such as data leakage protection [140], binary decision diagrams [139] and encryption [141], [142]. Finally, there are several secu-

rity solutions could be implemented to enhance the security of data transmission in IoT environments. Encryption and key management techniques are of great important security solutions, which can be further classified into heterogeneous ring signcryption [147], certificateless signcryption protocol [149], hash function algorithms [43], elliptic curve cryptography [48]–[50], [53], pseudonym based cryptography and identity based encryption [53]. Other security solutions could also be used to protect data transmission including quantum cryptography protocols (i.e. Quantum walk model) [163], Blockchain [162], Watermarking [160], machine learning techniques [146], trust model [52] and hardware-based solutions [150], [152], [153].

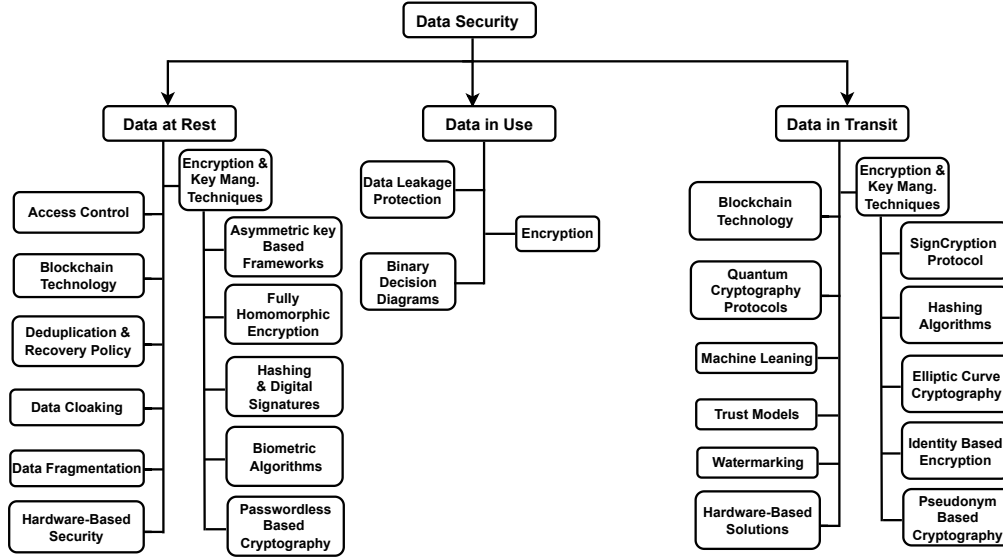


Figure 3.2: Taxonomy of Countermeasures Based on The Three Data States.

3.7 Concluding Remarks

- Protecting data at rest is of utmost importance for two primary reasons. Firstly, the constrained capabilities of IoT devices, such as limited memory capacity, computational power, and bandwidth, hinder the direct implementation of conventional security mechanisms, making it exceedingly difficult without adjustments. Secondly, there is a lack of established security guidelines and proper mitigation techniques for securing data at rest in IoT environments [103].
- Data in transit is particularly susceptible to security attacks compared to data in other states. In many cases, data is transmitted between IoT devices in plaintext, leaving confidential information in enterprise systems vulnerable. Consequently, it is crucial to protect data in transit from unauthorized access and possible threats that may jeopardize the privacy and integrity of the transferred information. Additionally, the diverse range of communication channels in IoT environments necessitates the use of various communication protocols and techniques, which may hinder the implementation of certain security solutions.
- It is important to note that no single countermeasure can provide absolute protection against cyber attacks. Therefore, adopting a multi-layered approach to security, regularly updating security measures, and staying informed about emerging threats are essential practices.

3.8 Future Directions

Several promising future research avenues should be investigated by IoT security society. This section suggests the following directions that could be pursued in the future:

1. Significant advancements have been made in developing security counter-measures for traditional computational and communication environments. However, applying these solutions in the context of IoT presents unique challenges due to the substantial differences in computation, storage, and communication capabilities between conventional and IoT devices and functionalities [22]–[24]. As a result, there is a growing need for lightweight and efficient schemes to ensure secure data transmission in IoT environments.
2. Developing and fine-tuning the internal structure of cryptographic algorithms to be suited for resource-constrained IoT devices. The development could be done for example by reducing key and block size, designing simple process to schedule the secret key, introducing simpler block and round structure,.etc.
3. Providing end-to-end mutual authentication and data integrity properties is of great essential to guarantee the utmost confidentiality of the sensitive data of IoT applications. This discipline is continually evolving, and new developments are being made all the time such as hybrid cryptographic solutions and emerging new biometric authentication techniques.

4. Machine learning is one of the revolutionary technologies of the modern day and becomes the most popular research topic. Additional research should be conducted to utilize machine learning principles to support the privacy and security for IoT systems. For instance, machine learning methodologies could be used to select an appropriate encryption technique dynamically based on the frequency with which the data being encrypted is accessed. Another future direction is to develop a predictive model (i.e., machine learning based IDS for IoT) to identify the common attacks in IoT environments such as MITM and DoS attacks.
5. To the extent of our knowledge, there is no common standard or methodology to assess the existing security solutions for data status. Therefore, it is recommended to propose a framework of security strategies and countermeasures to refine and optimize the design of IoT security solutions.

CHAPTER 4

CRYPTOGRAPHIC TECHNIQUES

Cryptography assumes a critical and indispensable role within the domain of IoT and is crucial for ensuring the security and privacy of interconnected devices and the generated data they produce. The significance of cryptography in IoT lies in its ability to safeguard confidential data from unauthorized access, tampering, and interception. The use of cryptographic techniques in IoT systems enables secure communication between devices, ensuring that data transmitted over networks remains confidential and cannot be easily deciphered by malicious actors. Encryption algorithms are employed to encode data, making it unintelligible to anyone without the necessary decryption keys.

By leveraging cryptography, IoT devices can authenticate and verify the identity of other devices, establishing trusted connections within the network. This authentication process helps prevent unauthorized devices from gaining access to

the system, safeguarding against potential security breaches. Additionally, cryptography provides integrity and non-repudiation in IoT environments. Through the use of digital signatures and hashing algorithms, data integrity can be ensured, detecting any unauthorized modifications or tampering. Digital signatures also enable the verification of the source of data, preventing the denial of data transmission or authentication attempts.

Overall, cryptography is indispensable in IoT as it addresses the critical security challenges faced by interconnected devices. By implementing robust cryptographic measures, organizations and individuals can protect sensitive information, maintain the trustworthiness of IoT systems, and mitigate risks associated with unauthorized access and data breaches. This chapter presents the basic concepts and preliminaries related to the modern cryptographic techniques such as Elliptic Curve Cryptography (ECC), Elliptic Curve Menezes-Qu-Vanstone (ECMQV), and Advanced Encryption Standard (AES).

4.1 Overview

Given the exponential expansion and advancement of internet-based applications and services, privacy becomes a primary concern. As the privacy of information and data is important to all organizations and business, cryptography is gaining the attention of researchers around the world.

Cryptography is defined as the science of secret writing. It is the study and practice of securing information or data by converting a private message into un-

readable format such that the message will be hidden from third parties and public readers. Cryptography satisfies many of security requirements including but not limited to confidentiality, authentication, integrity, and non-repudiation. Plain-text or clear text is data that can be understood, viewed, or read without using any special approaches or measures. Encryption is hiding plain-text such that it cannot be read or understood. Encrypting a plain-text results in an obscure semi-random text called cipher-text. Decryption involves the transformation of cipher-text into its original plain-text form, allowing for the recovery of the underlying message or information. A cryptosystem is a system that provides both encryption and decryption mechanisms [164].

Confidential information should be encrypted by the sender before transmission and decrypted by the receiver upon receipt using secret key(s). Modern cryptography can be divided into two types, namely: symmetric and asymmetric, based on the type of key(s). Symmetric cryptography involves encryption methods that utilize a single key for both the encryption and decryption processes, while asymmetric cryptography employs a pair of mathematically related keys for encryption and decryption purposes [164]. This distinction between symmetric and asymmetric cryptography lies at the core of secure communication and data protection.

In symmetric key algorithms, there are two main classifications: block ciphers and stream ciphers. Block ciphers operate on fixed-size blocks of data, encrypting and decrypting them as discrete units. Stream ciphers, on the other hand, encrypt and decrypt data in a continuous stream, typically one bit or one byte

at a time [164]. These classifications within symmetric key algorithms provide different approaches to secure data transmission and storage, allowing for flexibility in choosing the most appropriate method based on specific requirements and constraints. The following sections present the basic concepts and preliminaries related to AES, ECC and ECMQV.

4.2 Symmetric Cryptography

Symmetric cryptography can be defined as a cryptosystem scheme that consists of sets of transformations. The encryption transformation is defined as $E_e : e \in K$ and the decryption transformation is defined as $D_d : d \in K$, where K is the key space. In symmetric key encryption schemes it is computationally “easy” to find d by knowing e only and finding e knowing d . In most of the practical symmetric cryptosystem schemes we find that $e = d$, thus they are called symmetric-key cryptosystem schemes. In the literature, symmetric cryptosystems are referred to as private key, one-key, and single-key cryptosystem schemes. In symmetric cryptography, there are two main classifications: block ciphers and stream ciphers. Block ciphers encrypts blocks of data compared to individual characters in stream ciphers. One limitation of symmetric cryptosystems is the need for key exchange schemes [164].

4.2.1 Advanced Encryption Standard

AES is a block cipher scheme that was developed by the U.S. NIST in 2001. AES is based on a substitution and permutation design principle. AES functions by operating on data blocks of a standardized size of 128 bits. It offers support for key sizes of 128, 192, and 256 bits, which correspond to the number of bits in the encryption key used. The number of rounds, which determines the complexity of the encryption process, varies based on the size of the key utilized, with 10 rounds employed for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys. [164]. AES's establishment as a widely adopted block cipher has made it a cornerstone of modern cryptographic systems, ensuring secure and efficient data protection in various applications. As shown in Figure 4.1, the round structure of AES can be described as follows:

1. **Round Key Computation:** Using the cipher key, round keys are computed. The computations are based on Rijndael's key schedule [165]. Therefore, as the key size of AES increases more round keys can be generated, increasing by that the number of rounds [164].
2. **Add Round Key:** In this stage, a bit-wise XOR is performed on every byte with a block of the round key [164].
3. **Substitute Bytes:** A non linear look-up table is used to replace each byte by another [164].
4. **Shift Rows:** In this stage a circular left shift is used to shift the bytes in

each row. The rows following the first row, specifically the second, third, and fourth rows, undergo shifts with respective offsets of one, two, and three positions [164].

5. **Mix Columns:** Each column is multiplied by a static known matrix. The goal of this step is to use invertible linear transformation that provides diffusion in the cipher-text [164].

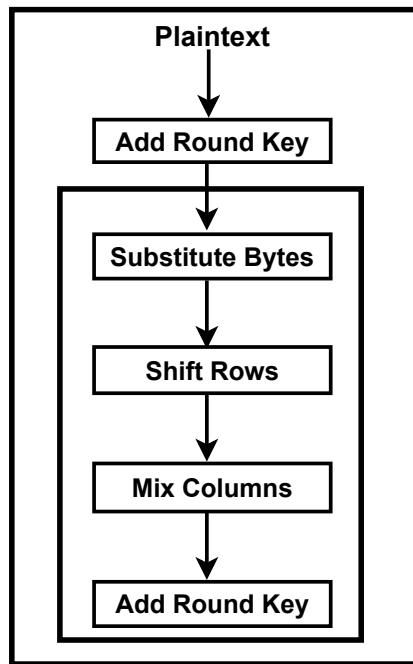


Figure 4.1: AES Round Structure

4.2.2 Limitations of Symmetric Key Cryptography

While symmetric key cryptography offers several advantages such as speed and efficiency [164], it also has certain limitations that can impact its applicability in certain scenarios. One major limitation is the key distribution problem. Since the same key is employed for both the encryption and decryption operations, it

needs to be securely shared between the communicating parties. This becomes challenging, especially in large-scale systems where numerous users or devices are involved. Establishing a secure and reliable key distribution mechanism can be complex and resource-intensive [96], [166].

Another limitation is the lack of inherent support for secure communication over an untrusted network. Symmetric key cryptography assumes that the communicating parties share a pre-established secret key [96], [164], [167], [168]. However, in scenarios where two parties have no prior communication or trust relationship, securely exchanging the secret key becomes a significant challenge. This limitation makes symmetric key cryptography less suitable for applications such as secure communication over the internet [167], [168].

Furthermore, the scalability of symmetric key cryptography is limited [169], [170]. In large-scale systems, where multiple users or devices need to communicate securely with each other, the number of keys required grows quadratically. Managing and securely storing a large number of keys can become impractical and resource-intensive, making symmetric key cryptography less efficient for such scenarios [96], [164].

These limitations have led to the development and widespread use of public key cryptography, interchangeably referred to as asymmetric key cryptography, which overcomes many of the challenges associated with symmetric key cryptography. Asymmetric key cryptography offers solutions to the key distribution problem and enables secure communication over untrusted networks, making it a more versatile choice in many modern applications [171].

4.3 Public Key Cryptography

Public key cryptography can be defined as cryptosystem schemes that consist of sets of transformations. The encryption transformation is defined as $E_e : e \in K$ and the decryption transformation is defined as $D_d : d \in K$, where K is the key space. In public key cryptosystem schemes, it is practically impossible, in terms of computational feasibility, to find d by knowing only e . Moreover, given a random cipher-text $c \in C$ and knowing E_e it is computationally infeasible to find $E_e(m) = c$. In most of the practical public-key encryption schemes we find that E_e is a trapdoor one-way function, where d is the information to compute the inverse function therefore allowing the decryption. In the literature, public key is referred to as asymmetric key, and two-key encryption schemes [164].

One major concern is the security level of public-key schemes. It is proven that only the one-time-pad -where the key size is at least as long as the message to be encrypted- is secure against any attack, given that the attacker has unlimited resources [164]. Unfortunately, there are no known cryptosystem that fulfill the one-time-pad property. Therefore, all public key cryptosystems are vulnerable to *brute-force attacks*. However, brute-force attacks are impractical if the computation power needed cannot be afforded by all potential attackers – named by Claude Shannon the “work factor”. Although, by simply increasing the key length -in many cases-, the work factor can be increased. In other algorithms, the work factors are very low, making them irrelevant to resist brute-force attacks. Moreover, specific algorithms have been developed to reduce the effort needed to

attack public key schemes. In other words, many public key schemes such as RSA and ElGamal are vulnerable to attacks that are much faster than the brute-force approach. Nowadays, with the dramatic decrease in the cost of computing power in addition to new mathematical breakthroughs, work factors have decreased dramatically. Some formerly promising public key algorithms have been found to be vulnerable after major weakness have been discovered. One example is the 'knapsack cryptosystem' algorithm, which was found to be vulnerable after the development of Shamir's polynomial time attack [172]. Moreover, some attacks target the implementation of a public-key algorithm and not the scheme itself. For example, side channel attacks target the implementation of an encryption scheme, and are based on measuring the duration required for an encryption hardware to transform plain-text into cipher-text, so that it becomes possible to expedite the search for potential decryption keys. Thus, by just using of asymmetric cryptosystem algorithms, security is not ensured [164].

4.3.1 Elliptic Curve Cryptography

Each public-key cryptosystem is constructed upon a challenging mathematical problem that renders the computation of the encryption key impractical or unfeasible. The strength of the corresponding cryptographic system relies on the relative complexity involved in solving that particular problem. The more challenging and computationally demanding the problem is, the stronger the system becomes in terms of its resistance against unauthorized access, tampering, or decryption [96], [164]. Examples of well-known public-key cryptosystems are Rivest-

Shamir-Adleman (RSA), ECC, ElGamal, Digital Signature Algorithm (DSA), and Diffie-Hellman. Compared with other well-known public-key cryptosystems, ECC is more suitable candidate due to its ability to offer high level of security with significantly smaller key sizes [173]–[176]. Figure 4.2 shows an example of adding two points (P,Q) on an elliptic curve which is the basic computations operation on ECC.

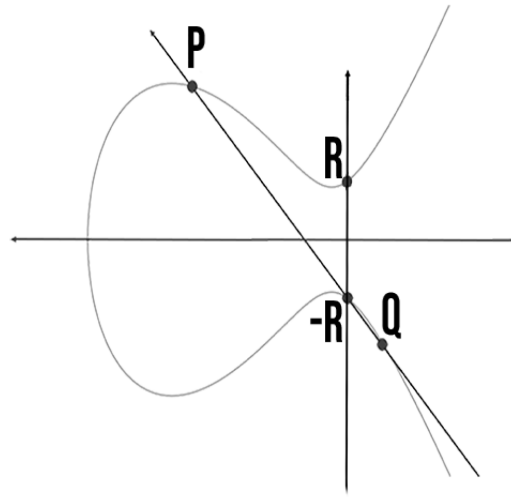


Figure 4.2: Point Addition on ECC [1]

Elliptic curves are defined as shown in Equation 4.1 where a_1, a_2, a_3, a_4, a_5 , and a_6 are coefficients; and x and y are variables.

$$y^2 + a_1xy + a_2y = a_3x^3 + a_4x^2 + a_5x + a_6 \quad (4.1)$$

In cryptography, finite fields are used to represent both variables and coefficients. ECC is implemented with a group of points that satisfy an elliptic curve equation defined over a finite field. The main cryptography operation in ECC is scalar

multiplication used to compute $Q = kP$, where P is a point on the elliptic curve multiplied by a scalar integer k resulting in a new point Q on the curve. To execute scalar multiplication, a combination of point addition and point doubling operations are employed. The main mathematical problem that ECC is based on is the *Elliptic Curve Discrete Logarithmic Problem (ECDLP)*. Therefore, the security strength of ECC depends on the hardness of ECDLP. ECDLP states that given P and $Q = kP$, it is mathematically infeasible to find k .

Every elliptic curve has two important parameters namely, the elliptic curve equation and the base point G , that is fixed for every curve. In elliptic curve cryptosystems, the private key is defined with a large random integer k while the base point for the curve G defines the corresponding public key [177].

4.3.2 Elliptic Curve Menezes-Qu-Vanstone

ECMQV, a key agreement protocol, has been standardized by Chinese standard GM/T 0009-2012 (SM2), IEEE 1363, ISO/IEC 15946-3, and ANSI X9.63 [56]. ECMQV is based on ECC and includes an implicit signature authentication function [178]. ECMQV is an authenticated version of the Elliptic-Curve Diffie-Hellman (ECDH) with protection against MITM attacks [179], [180]. ECMQV can be used as a substitute for the use of ECDHE+ECDSA-based authenticated key agreement technique [181]. Establishing a key agreement system using ECMQV produces the best outcomes in terms of security and efficiency [180]–[186]. Each entity in ECMQV protocol has both static and ephemeral private/public key pair. A shared key is then generated from the ephemeral and static keys, which ensures

generation of unique shared keys. The steps of ECMQV key agreement protocol are shown in Figure 4.3. For more details, please refer to Section 6.3.2.

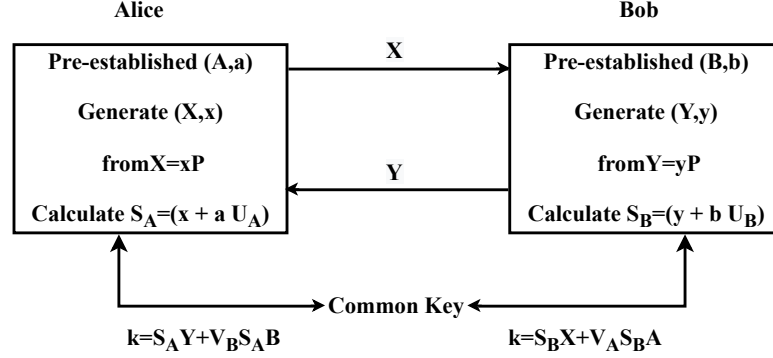


Figure 4.3: ECMQV Key Agreement Protocol

4.3.3 Limitations of Asymmetric Key Cryptography

While asymmetric key cryptography offers several advantages, it also has certain limitations that should be considered. One limitation is the computational overhead associated with asymmetric key operations. Asymmetric key algorithms, such as RSA or ECC, involve complex mathematical calculations that are computationally intensive compared to symmetric key algorithms. As a result, asymmetric key cryptography tends to be slower and requires more computational resources, making it less optimal for use in applications that demand real-time encryption and decryption [187], [188].

Another limitation is the reliance on a trusted public key infrastructure (PKI) for key management. The foundation of asymmetric key cryptography rests upon the concept of trust in public key certificates that are issued by a reputable certificate authority. However, the security of asymmetric key cryptography relies

heavily on the integrity and trustworthiness of the PKI [96], [164], [189]. If the PKI is compromised, an attacker can impersonate a legitimate entity or perform MITM attacks, undermining the security of the communication [96], [164], [190].

Additionally, asymmetric key cryptography exhibits reduced efficiency when it comes to encrypting substantial volumes of data in comparison to symmetric key cryptography. Asymmetric key algorithms have a significantly lower encryption throughput, and using them to encrypt large files or streams can be impractical and inefficient [191]–[193]. Therefore, a common practice is to use asymmetric key cryptography for securely exchanging a shared symmetric key, subsequently utilized for efficient bulk encryption of the actual data [193].

Despite these limitations, asymmetric key cryptography remains a crucial component of modern cryptography systems due to its unique capabilities, such as secure key exchange, digital signatures, and key authentication. By understanding its limitations and leveraging it in combination with symmetric key cryptography, practitioners can design secure and efficient cryptographic solutions for various applications [193].

CHAPTER 5

LITERATURE REVIEW

Over the past few years, there has been an increasing focus on advancing secure authentication methods tailored for IoT. A plethora of schemes have been suggested during this period to address the imperative need for ensuring secure and confidential data transmission within the IoT ecosystems. The rationale behind this is that IoT devices frequently establish connections with the internet, rendering them susceptible to an array of potential attacks. This chapter provides overview of efforts documented in various literature for different lightweight data transmission and authentication schemes have been put forth with the objective of fortifying the security of data transmission within IoT environments.

5.1 Non-ECC Based Schemes

A novel key management approach [156] was introduced to ensure the secrecy of data transmission in IoT-enabled WSNs. The scheme suggested to overcome the main security weakness of [157], which is the lack of session key agreement

and mutual authentication. However, performing identity authentication needs a significant amount of energy [158].

Authors in [159] proposed a technique to secure data transmission between a server, group of sensors, and a third party. The technique aims to reduce and improve sensor energy consumption. However, the proposed technique does not achieve mutual authentication. In addition, it has high communication overhead which renders it inefficient for WSN-based IoT applications.

An authentication scheme [52] was introduced to protect data transmission for heterogeneous devices in IoT environments. This scheme is based on the least trust-required model and utilizes the physical unclonable functions to simplify the suggested authentication system to mitigate cloning attacks. However, the computation and communication costs of the proposed scheme are high which make it inefficient for IoT environments [155].

Authors in [53] introduced a lightweight secure IoT scheme based on pseudonym based cryptography, identity based encryption and ECC. This approach facilitates both message communication and authentication between fog nodes and IoT devices. The scheme achieves authentication and deals with the intractability issue among IoT nodes. However, the computation cost of the scheme is high as it employs the pairing operation, which is a costly process [52], [53].

Furthermore, a new approach is proposed in [160] to create a data transmission and sinkhole detection model. The model is based on watermarking and homomorphic encryption methods. The watermarking technique is used to achieve data authentication whereas homomorphic encryption is used to provide efficient way

Table 5.1: Limitations of Exiting Non-ECC Based Schemes

Ref.	Limitations
[156]	Performing identity authentication needs a significant amount of energy [158]. High computation cost [52], [158].
[159]	Mutual authentication property is not achieved. Vulnerable to MITM attack. High communication cost.
[52]	High computation and communication costs [155].
[53]	High computation cost [52], [53].
[160]	Ineffective for detecting sinkhole attacks [161]. High communication and computation costs [194]–[196].
[42]	Computation cost should be considered to evaluate network performance [42].
[39]	Inefficient energy consumption [39].

to detect and prevent sinkhole attack. However, according to [161], the model proved ineffective for detecting sinkhole attacks in certain situations.

Li. et al. [39] proposed new approach to fortify the security of data transmission for WSNs. This approach is based on Paillier encryption algorithm and hash function. The proposed scheme applies deep learning techniques to accomplish efficient data sampling and compression. Then, Paillier encryption and hash function are used to achieve data integrity and confidentiality. The scheme needs more improvement to solve patio-temporal redundancy in order to enhance the scheme’s energy efficiency.

Authors in [42] proposed lightweight algorithm to enhance WSNs security by integrating two security protocols. The proposed algorithm achieves confidentially and integrity and enhances network’s energy cost. The study can be improved further, and the computation cost should be considered while evaluating the network performance [42]. Table 5.1 lists the non-ECC based related works discussed in this section and summarizes their limitations.

5.2 ECC Based Schemes

Various lightweight schemes have been introduced to fortify the security of data transmission in IoT environments. The authors in [48] proposed Secure Data Transmission Scheme (SDTS) in WSNs. The model consists of a base station, cluster heads and members. The scheme utilizes ECC to encrypt and decrypt transmitted data between IoT sensors. SDTS is resistant to replay and brute force attacks. However, the ephemeral public keys used to generate the shared key are susceptible to interception by any attacker, making them vulnerable to MITM attack [154]. In addition, the scheme does not achieve integrity and authentication between cluster heads and their members [154].

Another authentication scheme based on ECC and hash functions was proposed for WSN-based IoT systems [49]. The network model consists of gateway, users, and sensors. Users and sensors must authenticate themselves to the gateway before transmitting data between each other. This scheme is vulnerable to various attacks such as brute force and replay. In addition, the computation and communication costs of the scheme are high which render it inefficient for IoT environments [48].

Authors in [51] introduced another authentication scheme based on ECC and HTTP cookies to ensure the secrecy of transmitted data between embedded devices and cloud servers. The devices and cloud servers need to authenticate each other and agree on a secret key which is used to encrypt transmitted messages. This scheme fails to achieve integrity and mutual authentication [50], and has

high computation and communication costs [48].

An authentication scheme [50] was proposed for cloud servers and IoT environments. The scheme is composed of three phases, namely initialization, registration, login and authentication. The system parameters are set by cloud server. To register, each device needs to provide its ID and password. The cloud server then generates and sends a cookie to the embedded device. The scheme is based on ECC and it is designed to address security flaws in [51]. The scheme's major limitations are not ensuring data integrity and having high communication and computation costs [48].

In recent years, there has been an increasing focus on the advancement of secure authentication schemes specifically designed for the IoT. Dang et al. [197] proposed one of the recent secure authentication ECC-based schemes which offers efficient energy consumption and direct communications between devices. Another recent authentication scheme for IoT is the one proposed by Panda and Chattopadhyay [198]. This scheme uses ECC to provide secure mutual authentication between cloud servers and IoT devices. The implementation of this scheme is straightforward, making it highly accessible for deployment across a wide range of IoT applications. A novel, lightweight protocol has been introduced in [199] to enable end-to-end authentication using ECC. This meticulously designed protocol addresses and surpasses the inherent security vulnerabilities (i.e. forward secrecy, key control) of an earlier scheme proposed by Li et al. [200]. The Sowjanya et al. [199] protocol achieves much higher security standards in three well-defined phases, namely, Initialization, registration and authentication. The first phase en-

tails the definition of the system's parameters. In the next phase, each user needs to register with the network manager to receive authentication parameters for the system. In the third phase, the user's credentials are verified by the application server before being granted access to medical services. Sowjanya et al.'s protocol focuses on securing the communication between users and the application server using ECC. This advanced technique is employed to protect data transmission against cyber threats. The use of this technology coupled with the three-phase authentication process provides a robust framework for securing sensitive medical data.

In the realm of RFID security, Das et al. [201] developed a protocol for authentication that upholds the privacy of tags and enables mutual authentication of reader and tag. The approach involves using the public key of both components as their ID while keeping their private key concealed. The protocol is carried out in two phases. Firstly, during the setup phase, a one-time computation is carried out and stored for future use when tags are added or removed from the RFID system. Secondly, in the authentication phase, the reader and tag engage in on-demand communication. In a similar vein, Kalra and Sood [51] developed an authentication scheme based on ECC and HTTP cookies to ensure the secrecy of IoT systems. However, Chang et al. [202] identified two significant challenges associated with this approach: the attainment of mutual authentication and the generation of session keys were deemed unachievable. In an effort to address the shortcomings of Kalra and Sood's approach [51], Chang et al. [202] introduced an enhanced technique that sought to rectify its deficiencies. Nevertheless, it was

later revealed by Wang et al. [203] that the Chang et al. technique still lacked security, as an attacker could impersonate the server and establish a connection with a known device. A new approach was proposed in [203] to address security risks in the IoT network, allowing for secure communication between known devices and a server. This approach enables devices to communicate with a server without compromising their privacy, ensuring that the IoT ecosystem remains secure. By utilizing these methods, security protocols can be enhanced to prevent malicious actors from compromising the security of the IoT environment. The authors in [204] proposed a three-phase method to address privacy and information security concerns in IoT systems. In the first phase, to secure future communications, a distinct key is exchanged between child and parent nodes for encryption purposes. The second phase involves the encryption of data with different keys during intra-cluster communications, with updating the keys at each connection to ensure security. The third phase incorporates an authentication protocol for inter-cluster communications to prevent malicious nodes from joining the network. This serves as a safeguard to shield the network against unauthorized access and data tampering. Additionally, the authors demonstrated a notable enhancement in the network's performance through the implementation of the proposed method. Table 5.2 shows a summary of the ECC-based related works discussed in this section.

Table 5.2: Summary of Exiting ECC-Based Schemes

Ref.	Math. Model	Simulation	Performance Metrics
[48]	✓	✗	Communication, Computation costs
[49]	✓	✗	Communication, Computation, Storage costs
[50]	✓	✗	Communication, Computation, Storage costs
[51]	✓	✗	Communication, Computation, Storage costs
[197]	✓	✓	Computation, Storage costs
[198]	✓	✓	Communication, Computation, Storage costs
[199]	✓	✗	Communication, Computation, Storage costs
[200]	✓	✗	Computation cost
[201]	✓	✓	Communication, Computation costs
[202]	✗	✗	-
[203]	✓	✗	-
[204]	✗	✓	Computation cost

5.3 Lightweight ECC-Based Schemes Limitations

Table 5.3 lists the ECC-based related works discussed in Section 5.2 and summarizes their limitations. As shown in Table 5.3, several schemes have been proposed to fortify the security of data transmission in IoT [48]–[51], [197]–[201], [204]. However, they suffer from various security weaknesses such as MITM attack [48], [200], [201], brute force attack [49], [198]–[201], [204], and lack of integrity [48], [50], [51] and mutual authentication [48], [51]. Although some of the proposed schemes achieved various security requirements, their main drawback is high communication and computation costs [48], [197]–[200], [204]. This motivated us to propose a scheme to fill these gaps. As such, the proposed scheme will have the following properties: (a) ensures confidentiality, integrity, and end-to-end mutual authentication, (b) resists MITM, replay, and brute force attacks, and (c) is scalable and

lightweight in terms of communication and computation costs.

Table 5.3: Limitations of Existing Lightweight ECC-Based Schemes

Ref.	Limitations
[48]	Data integrity and mutual authentication are not achieved [154]. Subject to MITM attack [154].
[49]	Vulnerable to brute force and replay attacks [48]. High communication cost [48].
[50]	Data integrity is not achieved [48]. High communication and computation costs [48].
[51]	Data integrity and mutual authentication are not achieved [50]. High communication and computation costs [48].
[197]	High computation costs [197].
[198]	High communication and computation costs [198].
[199]	High communication and computation costs [199].
[200]	Subject to MITM attack [200]. High communication cost [200].
[201]	Subject to MITM attack [201].
[204]	High computation costs [204].

CHAPTER 6

PROPOSED LIGHTWEIGHT SECURE DATA TRANSMISSION SCHEME

A typical IoT ecosystem contains three layers namely, things, Fog, and Cloud. In the first layer, different data sources or things are connected to the Fog layer where more powerful nodes are available for data processing. On the other hand, the cloud layer offers a comprehensive and worldwide solution capable of managing vast volumes of data and expanding seamlessly through increased computing and storage capacity. To fully leverage the vision of IoT, the initial crucial step involves meticulously examining the data sources. These devices are employed to collect and distribute data, facilitating its widespread availability. Unfortunately, unsecure data dissemination will render IoT applications unreliable. Therefore, sensed as well as aggregated data must be disseminated in a secure fashion. Un-

fortunately and due to constrained resources in terms of storage, computing power, energy, and communications capabilities; lightweight secure and scalable dissemination schemes should be considered. This chapter describes the proposed LSDT scheme architecture and phases. In addition, this chapter discusses the trade-offs made as a result of the proposed scheme in different aspects, namely security, performance, resource utilization, and energy efficiency. Furthermore, this chapter presents the proposed scheme applications in wireless networks.

6.1 LSDT Architecture

WSNs are at the heart of IoT architectural implementation [205], [206]. WSNs, have played a pivotal role in the development of IoT technology which is proven to have a significant influence on the twenty-first century [207]–[210]. Because of their extensive usage in many vital and essential applications, WSNs have been regarded as one of the most important topics of study [156], [211]–[213]. Despite WSN-based IoT applications having significant use cases in intelligent environments, their adoption is difficult due to energy constraints in sensor nodes. WSNs' principal design goal is to maximize energy efficiency which can be achieved by reducing the computation and communication costs [205], [206], [214]. The constrained resources available in WSNs also make implementing security solutions challenging, making them vulnerable to cyber security threats [215].

Due to these reasons, this thesis proposes an LSDT scheme for WSNs. The network architecture of the proposed LSDT scheme shown in Figure 6.1 is com-

posed of the Base Station (BS) and the number of sensors. The *BS* is assumed to be a reliable, secure, and powerful computing device. The sensors are grouped into clusters to reduce the total power consumption of the network. Each cluster is represented by a Cluster Head (CH) and each sensor is referred to as a Cluster Member (CM). The *CMs* gather sensed data and transmit it to *CHs* using one-hop communication. The *CHs* gather the sensed data collected by *CMs* and forward it to the *BS*.

Each entity x in LSDT architecture has a static public key, namely K_x^{SPu} and static private key, namely K_x^{SPr} . Similarly, x has an ephemeral key pair, namely, (K_x^{EPu}, K_x^{EPr}) . In addition, two entities x and y can share a key represented as $K_{x,y}^{Sh}$. The architecture uses $C(m)$ to represent the ciphered message m while $V(z)$ is the Message Authentication Code (MAC) function output. Table 6.1 lists the notations used in the proposed scheme.

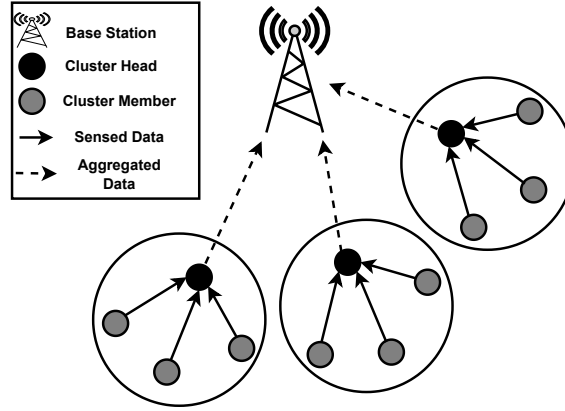


Figure 6.1: Network Architecture

Table 6.1: Proposed Scheme Notations

Notation	Description
p	Large prime number
$\mathbb{F}_p$	Prime field
G, q	Base point G of prime order q
K_x^{SPu}	Node x static public key
K_x^{SPr}	Node x static private key
K_x^{EPu}	Node x ephemeral public key
K_x^{EPr}	Node x ephemeral private key
s_A, s_B	Implicit signatures
$K_{x,y}^{Sh}$	Shared key between node x and node y
$K_{x,y}^{PSh}$	Pre-shared key between node x and node y
$CK_{x,y}^{Sh}$	Ciphered shared key between node x and node y
$C(m)$	Ciphered exchanged message m
$V(z)$	MAC function output
$E(K, m)$	Symmetric encryption of message m using key K
$D(K, m)$	Symmetric decryption of message m using key K

6.2 LSDT Overview

LSDT combines both symmetric and asymmetric protocols into one scheme to enhance performance and security as well as reduce the network's energy cost.

LSDT uses symmetric key-based protocol to ensure authentication between BS and CHs because of the following reasons: (a) symmetric key protocol consumes

less energy [184], [185], (b) although the communication cost is high in large networks [184], [185], this communication cost is minimized since the number of *CHs* are much smaller than *CMs*, and (c) processing time is minimal in symmetric key protocols since there is no need to perform any complex computations [184], [185]. This relieves *CHs* already overloaded with other tasks such as gathering and forwarding messages.

In addition, LSDT utilizes ECMQV, which is an asymmetric key protocol, to guarantee authentication between *CHs* and *CMs* for the following reasons. First, the ECMQV protocol is appropriate for a large number of sensors or when there is more than one-hop communication between *CMs* and *BS* [184], [185]. Second, the total amount of communication messages required for ECMQV is small [184], [185]. Specifically, authors in [216], [217] reported that ECMQV takes only two communication messages. Therefore, this improves the power consumption of the scheme. Finally, since ECMQV is lightweight [183]–[185], *CMs* are capable of performing ECMQV computation operations.

Using only symmetric-key protocols all the time has certain limitations that can impact its applicability in some scenarios. One major limitation is the key distribution problem. This becomes challenging, especially in large-scale systems where numerous users or devices are involved. Therefore, establishing a secure and reliable key distribution mechanism can be complex and resource-intensive [96], [166]. In addition, the scalability of symmetric key cryptography is limited. In large-scale systems, where multiple users or devices need to communicate securely with each other, the number of keys required grows quadratically in terms of the

number of users, while it grows linearly in case of asymmetric key. Managing and securely storing a large number of symmetric keys can become impractical and resource-intensive, making symmetric key cryptography less efficient for such scenarios [96], [164].

Furthermore, using only asymmetric-key protocol consistently will increase the computation cost, since asymmetric key computations are costly [187], [188]. Having a pair of public and private keys for each node is computationally intensive and requires more computational resources especially with CHs already overloaded with other tasks such as gathering and forwarding messages. That's why we propose a hybrid scheme that combines the security features of asymmetric-key protocols and the low cost of symmetric-key protocols.

6.3 LSDT Phases

To achieve its objectives, LSDT scheme utilizes three phases as shown in Figure 6.2. These phases are initialization phase, key establishment phase, and data transmission phase. It should be noted that the research community uses key generation and key establishment interchangeably.

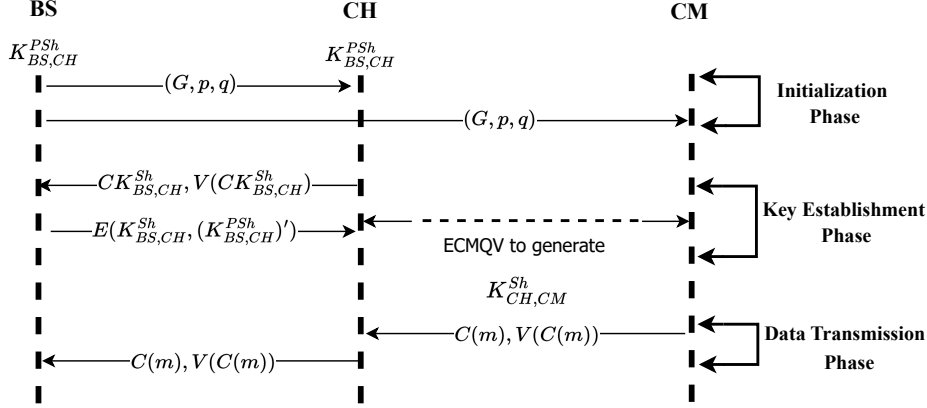


Figure 6.2: Proposed Scheme Phases

6.3.1 Initialization Phase

In this phase, we assume that a pre-shared key $K_{BS,CH}^{PSh}$ is established securely. In addition, the *BS* generates the scheme parameters including the elliptic curve $\mathbb{E}$ over $\mathbb{F}_p$, G, q , and publishes these parameters to *CH*s and *CM*s. We also assume that the *BS* sends the static private and public keys offline or through a secure channel to both *CM*s and *CH*s before their deployment into the system, which is common practice [201], [218], [219]. The public key, either static or ephemeral, is derived from its corresponding private key as shown in Equation 6.1.

$$K_x^{SPu} = K_x^{SPr} * G \quad (6.1)$$

Each *CM* and its *CH* also exchange their static public keys. This is needed for generating the shared key between *CM* and *CH* using ECMQV protocol during the next phase as described in Section 6.3.2. It should be noted that the initialization phase is only done once which aims just to generate and set the scheme

parameters. Hence, this phase is not included in the performance analysis of the scheme [48]–[50].

6.3.2 Key Establishment Phase Between CM and CH

Three steps are established in this phase. In the first step, the *CM* generates K_{CM}^{EPr} and then calculates K_{CM}^{EPu} . Similarly, the *CH* generates K_{CH}^{EPr} and calculates K_{CH}^{EPu} . In the second step, both *CM* and *CH* exchange their ephemeral public keys. Finally, the *CM* and *CH* need to generate the implicit signature to calculate the shared key as shown in Algorithm 1.

Algorithm 1 ECMQV Key Derivation for CM

Input: ECC parameters: $\mathbb{E}, p, G, q$. Private keys: $K_{CM}^{SPr}, K_{CM}^{EPu}$. Public keys: $K_{CH}^{SPu}, K_{CH}^{EPu}$
Output: Shared key $K_{CM,CH}^{Sh} \in \mathbb{E}(\mathbb{F}_p)$

1: $m \leftarrow \lceil \frac{(\log_2 q)}{2} \rceil$	▷ [m is the half bit length of q]
2: $u_A \leftarrow (u_x \bmod 2^m) + 2^m$	▷ [u_x is the x-coordinate of K_{CM}^{EPu}]
3: $s_A \leftarrow (K_{CM}^{EPr} + u_A \cdot K_{CM}^{SPr}) \bmod q$	▷ [Implicit Signature]
4: $v_A \leftarrow (v_x \bmod 2^m) + 2^m$	▷ [v_x is the x-coordinate of K_{CH}^{EPu}]
5: $z_A \leftarrow s_A \cdot v_B \bmod q$	
6: $K_{CM,CH}^{Sh} \leftarrow MPM(s_A \cdot K_{CH}^{EPu} + z_A \cdot K_{CH}^{SPu})$	

It should be noted that two scalar multiplications and a point addition are needed for a straightforward implementation of the Multiple Point Multiplication (MPM) operation (line 6 in Algorithm 1). However, Shamir’s method [220]–[223] allows to compute that with a cost close to one scalar multiplication as shown in Algorithm 2 and hence speeding up the computation process of the ECMQV protocol. This optimization is done by executing the elliptic scalar multiplication operations concurrently, especially with the pre-computed stage (line 1 in Algorithm 2). The identical bits of k and l are scanned from most significant to least significant bit. The intermediate value (line 2 in Algorithm 2), which is initially

set to infinity, is doubled for each bit as shown in line 4 in Algorithm 2. If the location of the scanned bit is $(k_i = 1, l_i = 0)$, $(k_i = 0, l_i = 1)$, or $(k_i = 1, l_i = 1)$, then P , Q , or $P + Q$ is added to the intermediate value, respectively as shown in lines 5 – 13 in Algorithm 2.

Algorithm 2 Multiple Point Multiplication (MPM)

Input: The points $P, Q \in \mathbb{E}(\mathbb{F}_p)$, two positive m -bit integers (scalars): $k = \sum_{i=0}^{m-1} 2^i k_i$, $l = \sum_{i=0}^{m-1} 2^i l_i$ where $k_i, l_i \in \{0, 1\}$ and $P = K_{CH}^{EPu}$, $Q = K_{CH}^{SPu}$, $k = s_A$, $l = z_A$

Output: $R = k \cdot P + l \cdot Q \in \mathbb{E}(\mathbb{F}_p)$

```

1:  $Z \leftarrow P + Q$  ▷ [Pre-computation Stage]
2:  $R \leftarrow O$ 
3: for  $i \leftarrow m - 1$  to 0 do
4:    $R \leftarrow R + R$  ▷ [Point Doubling]
5:   if  $(k_i = 1)$  and  $(l_i = 0)$  then
6:      $R \leftarrow R + P$ 
7:   end if
8:   if  $(k_i = 0)$  and  $(l_i = 1)$  then
9:      $R \leftarrow R + Q$ 
10:  end if
11:  if  $(k_i = 1)$  and  $(l_i = 1)$  then
12:     $R \leftarrow R + Z$ 
13:  end if
14: end for
15: return  $R$ 

```

6.3.3 Key Establishment Phase Between BS and CH

This phase is illustrated in Algorithm 3. Each CH generates $K_{BS,CH}^{Sh}$ and encrypts it using $K_{BS,CH}^{PSh}$ which is established securely between the BS and CH s during the initialization phase as shown in lines 2 and 3. In lines 4 – 6, CH sends the encrypted shared key $CK_{BS,CH}^{Sh}$ along with $V(K_{BS,CH}^{Sh})$ to BS , where $V(K_{BS,CH}^{Sh})$ is the output of hashing $CK_{BS,CH}^{Sh}$ with $K_{BS,CH}^{PSh}$ using the 256 hash function. Finally, CH cancels $K_{BS,CH}^{PSh}$ and a new pre-shared key is generated for the next round of the protocol to enhance the security of the scheme. It is noteworthy that the pre-shared key could be considered as a nonce in the proposed scheme in order to be resistant against replay attacks.

When the BS receives the data transmitted from CH , the BS calculates $(V(K_{BS,CH}^{Sh}))'$ and verifies the matching between the recalculated hash function and the received one $V(K_{BS,CH}^{Sh})$. If there is no matching, it rejects the message. Otherwise, it decrypts $CK_{BS,CH}^{Sh}$ using $K_{BS,CH}^{PSh}$ to get $K_{BS,CH}^{Sh}$ and saves it as a shared key between BS and CH as demonstrated in lines 9 – 12. The flow chart of this phase is illustrated in Figure 6.3.

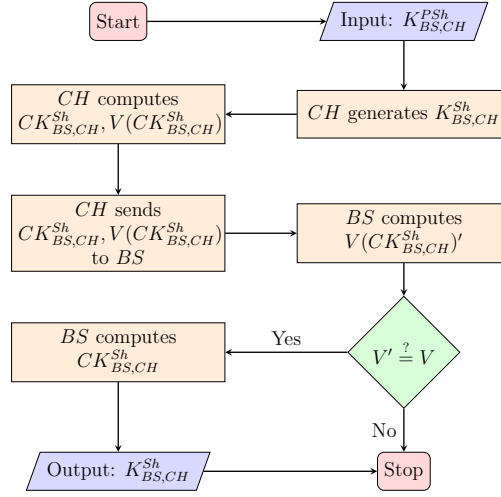


Figure 6.3: Key Establishment (CH-BS) Phase Flow Chart

Doing these steps guarantees establishing a shared key securely where only CH and BS have access to $K_{BS,CH}^{Sh}$ since it is encrypted by the pre-shared key which is known only to CH and BS . To verify the CH , the CH generates $K_{BS,CH}^{Sh}$ (as a nonce), encrypts it using $K_{BS,CH}^{PSh}$, and sends the message (encrypted nonce) to the BS . The BS decrypts the message and hence authenticates CH . To authenticate BS to CH , as shown in lines 13 – 14, the BS generates a new pre-shared key $(K_{BS,CH}^{PSh})'$ to be used for the next round. The new pre-shared key is sent to CH encrypted using $K_{BS,CH}^{Sh}$ which is known only to BS and CH . Therefore, $(K_{BS,CH}^{PSh})'$ can be accessed only by CH .

Algorithm 3 Key Establishment Between BS and CH

Input: $K_{BS,CH}^{PSh}$
Output: $K_{BS,CH}^{Sh}$ received by BS

- 1: **for** each CH **do**
- 2: Generate $K_{BS,CH}^{Sh}$
- 3: Compute $CK_{BS,CH}^{Sh} = E(K_{BS,CH}^{PSh}, K_{BS,CH}^{Sh})$ $\triangleright [Encrypt\ K_{BS,CH}^{Sh}\ using\ K_{BS,CH}^{PSh}]$
- 4: Compute $V(K_{BS,CH}^{Sh}) = MAC(K_{BS,CH}^{PSh}, CK_{BS,CH}^{Sh})$
- 5: Send $(CK_{BS,CH}^{Sh}, V(K_{BS,CH}^{Sh}))$ to BS
- 6: Cancel $K_{BS,CH}^{PSh}$
- 7: **end for**
- 8: **for** each $CK_{BS,CH}^{Sh}$ and $V(K_{BS,CH}^{Sh})$ received by BS **do**
- 9: Compute $(V(K_{BS,CH}^{Sh}))' = MAC(K_{BS,CH}^{PSh}, CK_{BS,CH}^{Sh})$
- 10: **if** $((V(K_{BS,CH}^{Sh}))' = V(K_{BS,CH}^{Sh}))$ **then**
- 11: Compute $K_{BS,CH}^{Sh} = D(K_{BS,CH}^{PSh}, CK_{BS,CH}^{Sh})$ $\triangleright [Decrypt\ CK_{BS,CH}^{Sh}\ using\ K_{BS,CH}^{PSh}]$
- 12: Keep $K_{BS,CH}^{Sh}$ in database as a shared key.
- 13: Generate $(K_{BS,CH}^{PSh})'$ $\triangleright [A\ new\ key\ for\ the\ next\ run]$
- 14: Send $E(K_{BS,CH}^{Sh}, (K_{BS,CH}^{PSh})')$ to CH $\triangleright [Encrypt\ (K_{BS,CH}^{PSh})'\ using\ K_{BS,CH}^{Sh}]$
- 15: **end if**
- 16: **end for**

6.3.4 Data Transmission Phase

The details of this phase are described in Algorithm 4. The data m sensed by the CM is encrypted using $K_{CM,CH}^{Sh}$ which is shared between each CM and its CH . The CM then sends the encrypted data $C(m)$ along with the resulting hash output $V(m)$ to the CH as shown in lines 1 – 5.

After receiving the encrypted message from its CM s, the CH checks if the recalculated hash function $(V(m))'$ matches the received hash function $V(m)$. If there is a match, it decrypts $C(m)$ using $K_{CM,CH}^{Sh}$ to get m . After gathering the message from all CM s, the CH aggregates and encrypts the sensed data using $K_{BS,CH}^{Sh}$. Finally, the CH sends the encrypted sensed data $C(m)$ along with $V(m)$ to the BS as demonstrated in lines 6 – 14.

Upon receiving $C(m)$ and $V(m)$, the BS verifies $(V(m))'$ and $V(m)$. If no match is found, the communication message is ignored. Otherwise, the BS de-

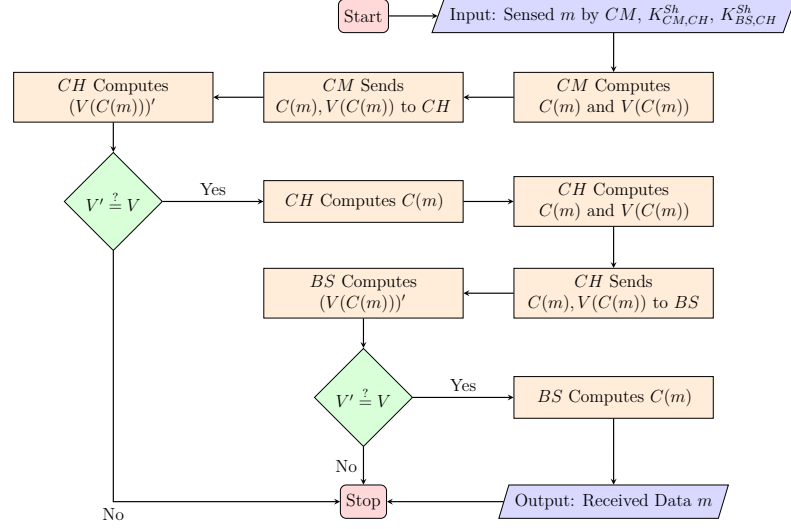


Figure 6.4: Data Transmission Phase Flow Chart

crypts $C(m)$ using $K_{BS,CH}^{Sh}$ to get m as shown in lines 15 – 20. The flow chart of this phase is highlighted in Figure 6.4.

Algorithm 4 Data Transmission Phase

Input: Sensed data m by CM , $K_{CM,CH}^{Sh}$ and $K_{BS,CH}^{Sh}$
Output: Received data m by BS

- 1: **for** each m sensed by CM **do**
- 2: Compute $C(m) = E(K_{CM,CH}^{Sh}, m)$ $\triangleright [Encrypt\ m\ using\ K_{CM,CH}^{Sh}]$
- 3: Compute $V(m) = MAC(K_{CM,CH}^{Sh}, C(m))$
- 4: Send $(C(m), V(m))$ to CH .
- 5: **end for**
- 6: **for** each $C(m)$ and $V(m)$ received by CH **do**
- 7: Compute $(V(m))' = MAC(K_{CM,CH}^{Sh}, C(m))$
- 8: **if** $((V(m))' = V(m))$ **then**
- 9: Compute $m = D(K_{CM,CH}^{Sh}, C(m))$ $\triangleright [Decrypt\ C(m)\ using\ K_{CM,CH}^{Sh}]$
- 10: Compute $C(m) = E(K_{BS,CH}^{Sh}, m)$ $\triangleright [Encrypt\ m\ using\ K_{BS,CH}^{Sh}]$
- 11: Compute $V(m) = MAC(K_{BS,CH}^{Sh}, C(m))$
- 12: Send $(C(m), V(m))$ to BS
- 13: **end if**
- 14: **end for**
- 15: **for** each $C(m)$ and $V(m)$ received by BS **do**
- 16: Compute $(V(m))' = MAC(K_{BS,CH}^{Sh}, C(m))$
- 17: **if** $((V(m))' = V(m))$ **then**
- 18: Compute $m = D(K_{BS,CH}^{Sh}, C(m))$ $\triangleright [Decrypt\ C(m)\ using\ K_{BS,CH}^{Sh}]$
- 19: **end if**
- 20: **end for**

6.4 The Trade-offs of LSDT Scheme

The proposed LSDT scheme combines symmetric and asymmetric protocols to enhance performance, security, and reduce energy costs. Let's discuss the trade-offs in different aspects:

- **Security:** The symmetric key-based protocol used for authentication between the BS and CHs may provide efficient authentication but lacks the strong security guarantees of asymmetric key protocols. Symmetric key protocols are vulnerable to key distribution issues, and if the key is compromised, it can lead to a complete security breach. On the other hand, the asymmetric key protocol, ECMQV, used for authentication between CHs and CMs, offers stronger security due to the use of public-private key pairs. This trade-off sacrifices some security strength in the BS-CH authentication for the efficiency and reduced communication overhead.
- **Performance:** Symmetric key protocols are known for their efficiency and low processing time since they do not involve complex computations. This makes them suitable for authentication between the BS and CHs, which are already burdened with other tasks. On the other hand, the asymmetric key protocol, ECMQV, requires more computational overhead due to the involved mathematical operations such as elliptic curve scalar multiplication. However, the total amount of communication messages required for ECMQV is reported to be small [184], [185], which helps reduce the overall communication cost.

- **Resource Utilization:** By using symmetric key-based authentication between the BS and CHs, the proposed scheme reduces the burden on CHs, which are already overloaded with tasks like message gathering and forwarding. This allows them to allocate their limited resources more efficiently. However, the use of the asymmetric key protocol, ECMQV, for CH-CM authentication may require additional computational resources on the CM side, as they need to perform ECMQV computation operations. The scheme assumes that CMs are capable of handling these operations efficiently. However, since ECMQV is lightweight [183]–[185], it is not a big burden on CMs to perform ECMQV computation operations since it is only done once during the key establishment phase.
- **Energy Efficiency:** The use of symmetric key-based authentication between the BS and CHs is justified based on the assumption that it consumes less energy compared to asymmetric key protocols. Symmetric key protocols typically involve simpler operations, requiring less computational effort and reducing energy consumption. This benefits the energy-constrained CHs. In addition, the smaller number of CHs compared to CMs helps minimize the communication cost of the proposed scheme.

In summary, the trade-offs made in the proposed LSDT scheme include sacrificing some security strength in the BS-CH authentication for efficiency and reduced communication overhead. The scheme leverages the energy efficiency of symmetric key protocols for BS-CH authentication, while using the stronger security guaran-

tees of the asymmetric key protocol, ECMQV, for CH-CM authentication. The trade-offs aim to optimize performance, security, resource utilization, and energy efficiency based on the specific requirements and constraints of the network.

6.5 LSDT Applications in Wireless Networks

The proposed LSDT scheme has potential applications in various wireless network scenarios where efficient data transfer, scalability, and energy savings are important considerations. Here are some use cases that demonstrate the applicability of the LSDT scheme:

- **IoT Networks:** In IoT networks, where a large number of sensors are deployed to collect data from the environment, LSDT can be beneficial. The use of cluster heads in LSDT allows for efficient data aggregation and transmission, reducing the number of direct sensor-to-base station communications. This helps in conserving energy and improving scalability, making it suitable for IoT deployments in smart cities, industrial monitoring, and environmental monitoring.
- **Wireless Networks:** WSNs consist of densely deployed sensor nodes that collaboratively monitor environmental conditions. The LSDT scheme can be applied in WSNs to improve data collection and transmission efficiency. By utilizing cluster heads, LSDT reduces the energy consumption of individual sensors and enables scalable network operation. This makes it applicable in

applications such as habitat monitoring, precision agriculture, and wildlife tracking.

- **Wireless Surveillance Systems:** In wireless surveillance systems, LSDT can enhance the scalability and efficiency of data transmission. By employing cluster heads, the scheme reduces the burden on individual surveillance cameras and facilitates aggregated data transfer to the base station. This can be useful in scenarios like large-scale video surveillance in public areas, where efficient utilization of bandwidth and energy is crucial.
- **Mobile Ad Hoc Networks (MANETs):** MANETs are self-configuring networks of mobile devices without a fixed infrastructure. The LSDT scheme can be adapted to MANETs to improve their scalability and energy efficiency. By introducing cluster heads, LSDT can help in organizing the network into clusters, reducing the overhead of node-to-node communication and facilitating efficient data transfer. This can be valuable in scenarios such as disaster response, military operations, and vehicular networks.
- **Wireless Sensor-Actuator Networks:** In networks where sensor nodes are interconnected with actuators to enable control and actuation of physical systems, LSDT can play a role in enhancing the network's performance. By utilizing cluster heads, the scheme reduces communication overhead and energy consumption, allowing efficient coordination between sensors and actuators. This can be applied in applications like industrial automation, smart buildings, and healthcare monitoring.

In summary, the proposed LSDT scheme is versatile and can be applied across various industries within the WSN-based IoT domains including e-healthcare systems, smart cities, and critical infrastructure systems like oil and gas supply, and power grids. The LSDT scheme addresses the crucial aspects of data security and authentication, which are integral to the seamless integration between IoT devices. Ensuring the security and integrity of data transmitted between IoT devices is of paramount importance in industries relying on IoT technology. By implementing LSDT scheme, organizations across diverse sectors can enhance the protection of sensitive data, establish secure communication channels, and authenticate the devices connected to the IoT network. For example, by applying LSDT scheme in e-healthcare systems, patients can establish secure and trustworthy communication channels with authorized doctors. Moreover, this scheme ensures the protection and confidentiality of all information exchanged between the patients and doctors, enhancing the overall security of the system. In conclusion, the LSDT scheme finds application in various wireless network scenarios where efficient data transfer, scalability, and energy savings are critical. Its use cases include IoT networks, wireless sensor networks, wireless surveillance systems, mobile ad hoc networks, and wireless sensor-actuator networks. By leveraging the benefits of cluster-based communication and reducing direct sensor-to-base station transmissions, LSDT offers improvements in performance, scalability, and energy efficiency in these applications.

CHAPTER 7

PERFORMANCE AND SECURITY ANALYSIS

This chapter evaluates the proposed LSDT scheme in terms of communication cost, computation cost, and security services. In addition, this chapter discusses the scalability and robustness of the proposed LSDT scheme.

7.1 Performance Analysis

In this section, the performance of LSDT scheme is evaluated in terms of communication and computation costs. As used by other researchers, The performance analysis utilizes 160-bit ECC [48]–[53], AES-128, and MAC-SHA [40], [224]–[230].

7.1.1 Communication Cost

The communication cost of the proposed scheme is equivalent to the total size of the scheme messages [48]–[51], [156], [198]–[201]. To calculate the size of the

transmitted messages during the scheme's phases, 160-bit ECC is used, where the ECC point $P = (x_P, y_P) \in E_p(a, b)$ is of size $(160 + 160) = 320$ bits. Furthermore, the sizes of the hash functions, random numbers, and symmetric encryption/decryption operations are also considered to be 160 bits. Therefore, the total communication cost (in bits) of the proposed scheme can be expressed as shown in Equation 7.1, where n is the number of transmitted messages. It is worth mentioning that we consider the communication cost for CM , CH , and BS for one data transmission [48]–[51], [156], [198]–[201].

$$\sum_{i=1}^n Message(i) \quad (7.1)$$

During the key establishment phase between CM and CH , the CM sends its ephemeral public key K_{CM}^{EPu} which is 160 bits long to the CH . The CH also sends its ephemeral public key K_{CH}^{EPu} which is 160 bits long to the CM . Therefor and as shown in Table 7.1, two messages are exchanged between CM and CH .

During the key establishment phase between CH and BS , the CH sends the encrypted shared key $CK_{BS,CH}^{Sh}$ along with $V(K_{BS,CH}^{Sh})$ to the BS ; (i.e. the message size is $160 + 160 = 320$ bits). The BS sends the encrypted new pre-shared key $(K_{BS,CH}^{PSh})'$ which is 160 bits long to the CH . These two messages exchanged between CH and BS are expressed in Table 7.1.

During the data transmission phase, the CM sends the encrypted data $C(m)$ along with the resulting hash output $V(m)$ to the CH ; (i.e. the message size is $160 + 160 = 320$ bits). The CH also sends its own $C(m)$ and $V(m)$ to BS (i.e.

the message size is $160 + 160 = 320$ bits). The total communication cost for one message transmission is $320 + 480 + 640 = 1440$ bits as shown in Table 7.1.

Table 7.1: LSDT Communication Cost

Scheme Phase	<i>No. of Messages</i>	<i>Cost(bits)</i>
Key Establishment (CM,CH)	2	320
Key Establishment (BS,CH)	2	480
Data Transmission	2	640
Total	6	1440

7.1.2 Computation Cost

Computation cost refers to the cost used to calculate the encryption and authentication operations of the scheme. The calculation method for computation cost involves a number of ECC point multiplications (T_m) and hash function operations (T_h) performed during the scheme's phases. As reported in the literature [48]–[52], [204], the computation cost of inexpensive operations (i.e., concatenation, comparison, XOR) is ignored. These studies consider only T_m and T_h to calculate the computation cost. They also demonstrated that $T_m \gg T_h$ which is shown by [204], [231] to be $7.3529ms$ for T_m and $0.0004ms$ for T_h . It is noteworthy that the computation cost for CM , CH , and BS is also computed for one data transmission [48]–[51], [156], [197]–[201], [204].

During the key establishment phase between CM and CH , as shown in Algorithm 1, CM calculates its ephemeral public key K_{CM}^{EPu} which takes one T_m . Then, in Algorithm 2, CM takes another T_m to calculate the shared key $K_{CM,CH}^{Sh}$. Sim-

ilarly, CH calculates its ephemeral public key $K_{CH}^{EP_u}$ taking one T_m and another T_m [216] to calculate the shared key $K_{CM,CH}^{Sh}$.

During the key establishment phase between CH and BS , as demonstrated in Algorithm 3, CH calculates $V(K_{BS,CH}^{Sh})$ taking one T_h . Similarly, BS calculates $(V(K_{BS,CH}^{Sh}))'$ taking also one T_h .

Table 7.2: LSDT Computation Cost

Scheme Phase	CM	CH	BS	<i>Total</i>
Key Establishment (CM,CH)	$2T_m$	$2T_m$	-	$4T_m$
Key Establishment (BS,CH)	-	T_h	T_h	$2T_h$
Data Transmission	T_h	$2T_h$	T_h	$4T_h$
Total	$2T_m + T_h$	$2T_m + 3T_h$	$2T_h$	$4T_m + 6T_h$

During the data transmission phase, CM calculates $V(m)$ (i.e., one T_h). After receiving the data from its CMs , the CH recalculates $(V(m))'$ taking one T_h . After gathering the data from all CMs , CH calculates $V(m)$ taking another T_h . Upon receiving the data, the BS recalculates $(V(m))'$ taking one T_h . Therefore, the total computation cost for one data transmission is $4T_m + 6T_h$ as shown in Table 7.2.

7.2 Security Analysis

This section states the security claims that are ensured by the proposed LSDT and provides the justification for each claim.

- **Claim:** LSDT ensures confidentiality.

Justification: The data is encrypted using 160-bit ECC and AES-128.

Even if an attacker is able to crack the secret key and asymmetric key pair in one round, the attacker cannot benefit from those keys since new keys are generated every round.

- **Claim:** LSDT ensures integrity.

Justification: Since nodes send encrypted messages as well as hashed messages, the receiving nodes can check that the message has not been modified or altered by verifying the hashed message. It should be noted that the shared key (i.e., either $K_{BS,CH}^{Sh}$, $K_{CH,CM}^{Sh}$) is used for encryption and hashing. Please refer to Sections 6.3.3 and 6.3.4.

- **Claim:** LSDT ensures end-to-end mutual authentication.

Justification: As mentioned in Section 6.3.1, both CH and BS have $K_{BS,CH}^{PSh}$. To verify the CH , the CH generates $K_{BS,CH}^{Sh}$ (i.e., a nonce), encrypts it using $K_{BS,CH}^{PSh}$, and sends the message (encrypted nonce) to the BS . The BS decrypts the message and hence authenticates CH . CH can authenticate BS in a similar manner. Furthermore, and since LSDT utilizes ECMQV between CH and its CMs , mutual authentication between CH and its CMs is ensured since it is inherited from ECMQV [179]–[181].

- **Claim:** LSDT resists the MITM attack.

Justification: Any attempt to launch an MITM attack is thwarted because the LSDT scheme achieves end-to-end mutual authentication as described

above.

- **Claim:** LSDT resists replay attack.

Justification: During the key establishment phase between *BS* and *CH*, each message includes an encrypted nonce to disable any attempt to perform a replay attack. If the attacker re-transmits a message using the same encrypted nonce, the receiving node can detect the replayed message immediately since a new key is generated in each round.

- **Claim:** LSDT resists brute force attack.

Justification: Since LSDT uses 160-bit ECC and AES-128, LSDT resists brute force attacks. This property is inherited from 160 bits ECC and AES-128 [232]–[234]. The key spaces for searching for the keys in 160-bit ECC and AES-128 using brute force attacks are 2^{160} and 2^{128} , respectively [48], [235]. In addition, the generated keys are used to encrypt/decrypt messages for only one round. As such, compromising the keys of a single round does not affect the overall security of the system.

7.3 Scalability and Robustness of LSDT

7.3.1 LSDT Scalability

LSDT is a cluster-based scheme, which is not only more efficient than multi-hop but more scalable as well. The use of cluster heads reduces the number of sensors that need to communicate with the base station, which saves energy. Therefore,

this allows the system to cover large areas with less energy. Another factor that may affect the scalability is the number of sensors in the network. LSDT uses ECMQV protocol which is suitable for large-scale sensor networks [184], [185]. In addition, processing time is minimal in symmetric key protocols, therefore, no need to perform any complex computations [184], [185]. Overall, it is worth mentioning that the best data transmission scheme will depend on the number of nodes and the specific nature of the application.

For a given application, with a specified number of nodes, n , the scalability of the proposed scheme is achieved by dividing the nodes into small clusters. Each node in the cluster is communicating with the cluster head. The cluster head is communicating with the base station. However, none of the other nodes communicates directly with the base station, except through the cluster head.

The following setup for LSDT is proposed. As the number of nodes n increases, more clusters will be created and the maximum size of the clusters is slightly increased to accommodate the increasing number of nodes in the scheme. Let n be the number of nodes; then the number of clusters is set to $c = \lceil \sqrt{n} \rceil$, and the maximum size of the clusters is given by:

$$max_size = \left\lceil \frac{n}{c} \right\rceil$$

For example, suppose that the scheme has $n = 110$ nodes. Therefore, the nodes are divided into $c = \lceil \sqrt{110} \rceil = 11$ clusters. Each cluster has at most $\lceil 110/11 \rceil = 10$ nodes, including a cluster head that communicates with all other

nodes in the cluster and the base station. So, the cluster head is communicating with at most $max_size + 1$ entities at any given time. If the number of nodes is increased to $n = 10000$, the scheme will have 100 clusters, with 100 nodes in each cluster, and so on.

Theorem 7.1 *The setup proposed for LSDT allows quadratic scalability of the number of nodes n in terms of both number of clusters c and their maximum size.*

Proof. By construction, we have $c = \lceil \sqrt{n} \rceil$, which implies that $max_size = \lceil n/c \rceil = \lceil n/\lceil \sqrt{n} \rceil \rceil \approx \sqrt{n}$. Therefore, the growth complexity of both c and max_size is $\Theta(n^2)$. Hence, the number of clusters and their maximum size increase linearly with quadratic growth in the number of nodes.

I

For example, suppose that we have 10 clusters, with 10 nodes each, we will have a total of 100 nodes (CMs). Then, suppose we increase the number of clusters linearly from 10 to 20 and the size of the cluster from 10 to 20, then the total number of nodes (CMs) will increase quadratically from 100 to 400.

In summary, the proposed LSDT scheme is designed as a cluster-based scheme, which offers advantages in terms of scalability and efficiency compared to multi-hop approaches. By introducing CHs, LSDT reduces the number of sensors directly communicating with the BS, resulting in energy savings and increased scalability. Instead of each sensor directly transmitting data to the BS, they communicate with the CHs, which then aggregate and transmit the data to the BS. This hierarchical structure enables the system to cover large areas while maintaining

efficiency. The proposed setup for LSDT aims to achieve quadratic scalability of the number of nodes (n) in terms of both number of clusters (c) and their maximum size. This scalability property suggests that as the network grows, the performance and efficiency of the scheme can be maintained.

7.3.2 LSDT Robustness

The introduction of CHs in the LSDT scheme can enhance the robustness of the network. CHs act as intermediaries between the sensors and the BS, aggregating and forwarding data. This can help in reducing communication overhead and improving network reliability. In case of node failures or communication disruptions, the CHs can still maintain connectivity with the sensors within their cluster, enabling local data aggregation and reducing the impact of individual node failures on the overall network performance.

7.3.3 Scalability Limitations

While the proposed LSDT scheme aims to achieve scalability benefits, it is important to consider the following potential limitations:

- **Cluster Size and CH Overhead:** As the number of clusters (c) increases, each cluster may become smaller in size, potentially leading to increased overhead due to the CHs' management and communication tasks. This overhead can affect the overall efficiency and scalability of the system.
- **Communication Bottlenecks:** Although LSDT reduces the number of

sensors directly communicating with the BS, the communication between CHs and the BS can become a potential bottleneck as the network scales. If the number of CHs is limited and the communication load increases significantly, it may lead to congestion or increased latency in transmitting aggregated data from the CHs to the BS. This can impact the overall scalability and performance of the scheme.

- **Dynamic Network Conditions:** The proposed LSDT scheme assumes a static cluster formation, where CHs and clusters are predetermined. However, in dynamic network scenarios where nodes join or leave the network frequently, maintaining the predefined clusters and ensuring optimal cluster formation can be challenging. This can limit the scalability of the scheme, as the network may require frequent reconfiguration and redistribution of nodes among clusters to adapt to changing conditions.

In summary, the LSDT scheme offers scalability benefits through the use of CHs and cluster-based communication. It aims to cover large areas with reduced energy consumption and achieve quadratic scalability. However, limitations such as cluster size and CH overhead, communication bottlenecks, and adaptability to dynamic network conditions need to be considered to ensure the scheme's scalability and robustness in practical deployment scenarios.

7.4 Comparative Analysis

This section presents the comparative analysis of LSDT scheme with other relevant security schemes.

7.4.1 Security Comparative Analysis

Security analysis of some security schemes [48]–[51] has been published in [48] as shown in Table 7.3. Authors in [48], examined security properties namely, confidentiality, integrity, and mutual authentication as well as schemes resistance to the common authentication attacks such as MITM, replay, and brute force attacks. Table 7.3 was discussed in details in [48] to conclude that some schemes are vulnerable to attacks and do not achieve some security properties such as data integrity [48], [50], [51] and mutual authentication [48], [51].

Table 7.3: Comparative Study on Security Services

Property	[48]	[49]	[50]	[51]
Confidentiality	✓	✓	✓	✓
Integrity	✗	✓	✗	✗
Mutual Authentication	✗	✓	✓	✗
MITM Attack Resistance	✗	✓	✓	✓
Brute Force Attack Resistance	✓	✗	✓	✓
Replay Attack Resistance	✓	✗	✓	✓

We were motivated by the work done in [48] to extend Table 7.3 and include more recently published security schemes as well as our own proposed LSDT scheme. Table 7.4 shows the comparison of these existing security schemes in terms of security properties and their resistance to different attacks. Although other schemes violate some security properties and are vulnerable to some attacks, it is observed that LSDT ensures security properties and resists different attacks.

Table 7.4: Security Services Comparison

Property	<i>LSDT</i>	[48]	[49]	[50]	[51]	[197]	[198]	[199]	[200]	[201]	[156]	[204]
Confidentiality	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Integrity	✓	✗	✓	✗	✗	✓	✓	✓	✓	✓	✓	✓
Mutual Authentication	✓	✗	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓
MITM Attack Resistance	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
Brute Force Attack Resistance	✓	✓	✗	✓	✓	✓	✗	✗	✗	✗	✗	✗
Replay Attack Resistance	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓

7.4.2 Performance Comparative Analysis

This section compares the performance of LSDT scheme with other relevant schemes in terms of communication and computation costs. In this section, we analyze and provide a comprehensive breakdown of communication costs associated with the existing related schemes to investigate the cost distribution among their phases as shown in Table 7.5. Upon careful observation, it is noticed that the proposed LSDT scheme exhibits comparable or lower communication costs during the key generation phase when compared to [49], [156], [198]–[200], and demonstrates competitive value in comparison to other schemes. It is also noticed

that LSDT scheme incurs lower communication cost during the data transmission phase, which is executed more often compared to the key generation phase, outperforming other schemes in this regard.

Table 7.5: Comparative Analysis of Communication Costs

Scheme	Key Gen.		Data Trans.		Total	
	# Messages	# Bits	# Messages	# Bits	# Messages	# Bits
[48]	3	480	2	1280	5	1760
[49]	4	2080	4	3680	8	5760
[50]	2	640	3	1760	5	2400
[51]	3	480	2	1760	5	2240
[156]	5	2400	3	2400	8	4800
[197]	-	-	-	-	-	-
[198]	2	800	3	1760	5	2560
[199]	2	1120	2	1440	4	2560
[200]	2	1120	2	960	4	2080
[201]	2	640	3	1120	5	1760
[204]	-	-	-	-	-	-
LSDT	4	800	2	640	6	1440

Furthermore, an in-depth investigation is conducted and a thorough analysis of computation costs associated with the existing related schemes is offered, aiming to explore how costs are distributed across their stages, as illustrated in Table 7.6. It is noticed that the proposed LSDT scheme demonstrates compara-

ble or lower computation costs during the key generation phase when compared to [156], [197], [199] while it showcases competitive value when compared to other schemes. Additionally, it is observed that LSDT scheme incurs lower computation cost during the data transmission phase, which occurs more frequently than the key generation phase.

Table 7.6: Comparative Analysis of Computation Costs

Scheme	Computation Cost		
	Key Gen.	Data Trans.	Total
[48]	$3T_m$	$2T_m + 2T_h$	$5T_m + 2T_h$
[49]	$25T_h$	$4T_m + 31T_h$	$4T_m + 56T_h$
[50]	$2T_m + 5T_h$	$8T_m + 7T_h$	$10T_m + 12T_h$
[51]	$2T_m + 3T_h$	$7T_m + 7T_h$	$9T_m + 10T_h$
[156]	$4T_m + 2T_h$	$7T_m$	$11T_m + 2T_h$
[197]	$8T_m + 9T_h$	$7T_m + 9T_h$	$15T_m + 18T_h$
[198]	$3T_m + 4T_h$	$8T_m + 9T_h$	$11T_m + 13T_h$
[199]	$6T_m + 2T_h$	$9T_m + 4T_h$	$15T_m + 6T_h$
[200]	T_m	$5T_m + 5T_h$	$6T_m + 5T_h$
[201]	$2T_m$	$4T_m + 4T_h$	$6T_m + 4T_h$
[204]	$2T_m + 3T_h$	$15T_m + 13T_h$	$17T_m + 16T_h$
LSDT	$4T_m + 2T_h$	$4T_h$	$4T_m + 6T_h$

CHAPTER 8

PERFORMANCE EVALUATION

This chapter presents the performance evaluation of the proposed LSDT scheme in terms of communication and computation costs. We establish the experimental environment. Then, we execute the experiments, gather the resulting data, and conduct a comprehensive analysis. Section 8.1 provides detailed information on the construction of the experimental platform, while section 8.2 outlines the approach taken in conducting our experiments. Section 8.3 discusses the experiment results and compares the performance of LSDT scheme with other relevant security schemes in terms of communication and computation costs.

8.1 Simulation Environment

In recent years, numerous IoT simulators have emerged. However, due to its widespread acceptance and popularity as an IoT operating system, we chose to

utilize Contiki [236] and its built-in Cooja simulator [237]–[239]. This section first provides a brief overview of the rationale behind selecting Contiki and Cooja. Then, we summarize the configuration settings and their corresponding values used throughout our experiments in Table 8.1.

8.1.1 Contiki Operating System

Contiki OS is an adaptable, lightweight, and highly portable operating system designed specifically for IoT. It serves as a versatile toolbox for organizing complex wireless systems. Contiki offers an extensive range of features, some of which are highlighted below:

- Contiki includes Cooja, a built-in emulator that provides hardware-level flexibility, allowing for versatile testing and analysis.
- Contiki incorporates dynamic and efficient memory management techniques, allowing it to function effectively with limited memory resources. It is capable of running on systems with small memory sizes, such as 10Kbytes of RAM and 30Kbytes of ROM.
- It supports a wide array of platforms including SKY, Wismote, and Z1 motes. It also caters to diverse microcontrollers such as TI MSP430 family, Atmel AVR family, and ST STM32W. Additionally, it supports various radios such as Texas Instruments CC2520, CC2420, and CC1020 .
- It provides support for two types of networking: IP networking and non-IP

networking. IP networking is enabled by the uIPv4 and uIPv6 stacks, while non-IP networking is facilitated through the Rime stack.

- Contiki implements the adaptation layer known as 6LoWPAN, which enables seamless operation with uIPv6.

8.1.2 Cooja Simulator

Cooja is a versatile simulator specifically developed for IoT nodes that operate on Contiki OS. It enables simulations of IoT nodes at different levels, including the machine code instruction set level, OS level, and network level. At the machine code instruction set level, Cooja employs MSPSim to emulate the IoT nodes, utilizing compiled Contiki C code tailored to the specific hardware device. On the other hand, at the OS level, the code of the nodes is directly executed in Contiki C on the host machine, disregarding the hardware limitations of the physical sensor nodes. This allows for a more flexible and convenient development environment, decoupled from the constraints imposed by the actual sensor hardware. Furthermore, at the network level, Cooja utilizes Java to simulate IoT nodes, allowing for abstract and high-level operations such as packet transmission and reception. The key features of Cooja can be summarized as follows:

- Cooja simplifies the process of developing and evaluating IoT network protocols by providing support for both graphical interfaces and command line.
- Simulations are conducted at a hardware level, employing genuine hardware profiles, which enables the compiled code to be seamlessly uploaded and

executed on real platforms. This simulation approach ensures that the code operates flawlessly and accurately on actual hardware, facilitating reliable deployment and testing in real-world scenarios.

- By accurately representing the instruction sets of IoT nodes, Cooja facilitates precise modeling of their intricate behavior at a granular level. This capability allows for a detailed and accurate simulation of how the nodes function, capturing the nuances of their operations with high fidelity.
- Cooja facilitates the evaluation of proposed solutions under realistic conditions by leveraging network models like topology, medium interference, and radio propagation.

Table 8.1: Simulation Configuration

Parameter	Value
Platform	Contiki OS v3.0
Simulator	Cooja
Network Topology	Random
Mote Type	Skymote with 10KB RAM and 48KB ROM
Network Layer	RPL + uIPv6 + 6LoWPAN
Radio Environment	Unit Disk Graph Medium (UDGM)
Transmission Range	50 meters
MAC & PHY	IEEE 802.15.4
Time Library	rtimer
Cryptographic Algorithms	<i>ECC</i> –160, <i>AES</i> –128, <i>MAC</i> –256

8.2 Experimental Methodology

Throughout our experiments, we employ the Cooja simulator to choose a suitable radio medium model, such as MRM, DGRM, or UDGM. Additionally, we select a specific mote type from the available options, including SKY mote, Wismote, or Z1. We then proceed to select a network topology, such as random positioning or a uniform 2D-Grid, followed by determining the transmission range for the populated nodes. It should be noted that all Contiki default settings are used unless otherwise specified. To measure the computation cost, the rtimer library of Contiki is used [228], [236]. The experimental process phases are depicted in Figure 8.1, providing a visual representation of the steps involved in conducting our experiments.

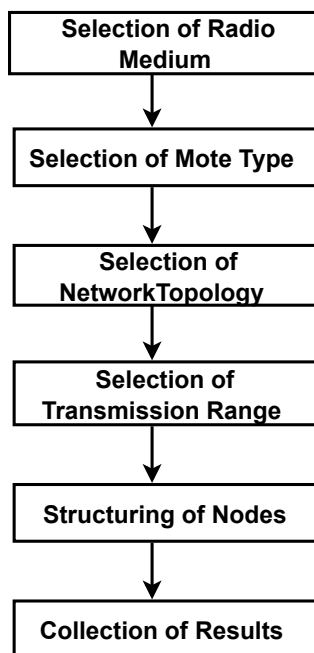


Figure 8.1: Experimental Process Approach

The architecture of our proposed scheme is composed of base station and number of sensors. The sensors are grouped into clusters in order to reduce the total power consumption of the network. Each cluster is represented by cluster head and each sensor is referred to as cluster member. For our experimental purposes, three clusters are organized, with each cluster consisting of one cluster head and three cluster members. The network topology is shown in Figure 8.2 where nodes 1 – 6 are cluster members, nodes 7 and 8 represent cluster head and node 9 is a base station.

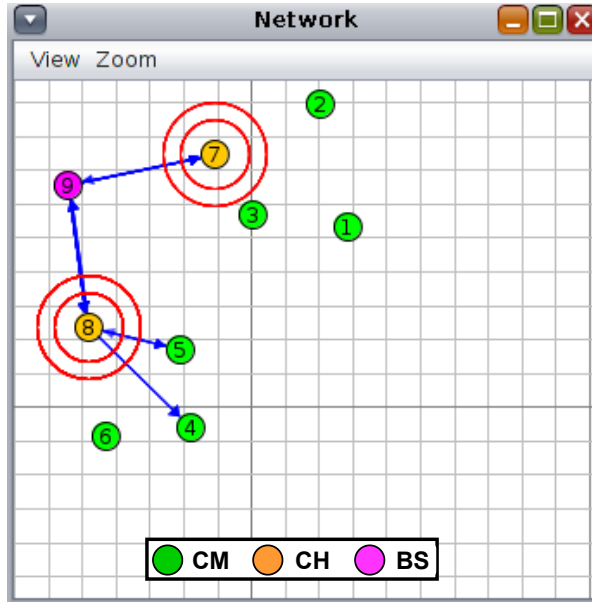


Figure 8.2: Network Topology

8.3 Simulation Results and Discussion

The objective of the conducted experiments is to evaluate the performance of the proposed LSDT scheme. It is worth mentioning that the Skymote platform is used for conducting these experiments. Each ECC multiplication operation

(T_m) takes $2.93s$, and the MAC hash function (T_h) operation takes $0.06s$. The total computation time for one data transmission is $4(2.93) + 6(0.06) = 12.1s$ as shown in Table 8.2. It is observed that during the key establishment phase between CM and CH, the CM incurs a computation cost of $2T_m = 5.86$, indicating that it involves two T_m operations with a duration of $2.93s$ each. Similarly, the CH also incurs a computation cost of $2T_m = 5.86$, representing two ECC point multiplication operations. Both the CM and CH require two T_m operations each to compute their ephemeral public keys and the shared key between them, as elaborated in Section 7.1.2. The BS does not contribute to the computation cost in this phase, as indicated by the "-". The total computation cost for this phase is $4T_m = 11.72$. During the key establishment phase between CH and BS, the CM does not contribute to the computation cost, as indicated by the "-". The CH incurs a computation cost of $T_h = 0.06$, representing one T_h operation with a duration of $0.06s$. Similarly, the BS also incurs a computation cost of $T_h = 0.06$, indicating one hash function operation. The total computation cost for this phase is $2T_h = 0.13$. In the data transmission phase, the CM incurs a computation cost of $T_h = 0.06$, representing one hash function operation. The CH incurs a computation cost of $2T_h = 0.13$, indicating two hash function operations. The BS also incurs a computation cost of $T_h = 0.06$, representing one hash function operation. The total computation cost for this phase is $4T_h = 0.26$. Please refer to Section 7.1.2 which offers a meticulous breakdown and analysis of the cost components.

Table 8.2: LSDT Computational Time

Scheme Phase	<i>CM</i>	<i>CH</i>	<i>BS</i>	<i>Total</i>
Key Establishment (CM,CH)	5.86	5.86	-	11.72
Key Establishment (BS,CH)	-	0.06	0.06	0.13
Data Transmission	0.06	0.13	0.06	0.26
Total	5.92	6.05	0.13	12.1

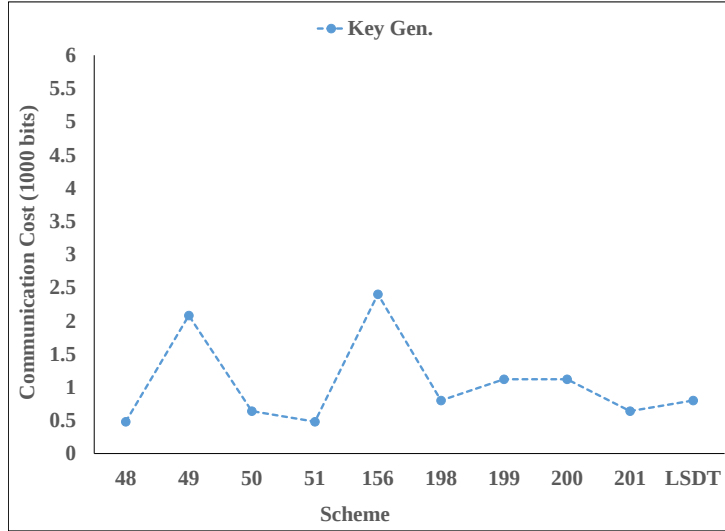


Figure 8.3: Communication Cost - Key Establishment Phase

Through careful observation, during the key generation phase, it is noticed that the proposed LSDT scheme demonstrates communication cost that is either comparable to or lower than those of [49], [156], [198]–[200]. Furthermore, when compared to other schemes, the LSDT scheme proves to be highly competitive in terms of its overall value as shown in Figure 8.3. During the data transmission phase, which occurs more frequently than the key generation phase, Figure 8.4

demonstrates that the LSDT scheme incurs significantly lower communication costs, surpassing the performance of other schemes in this aspect.

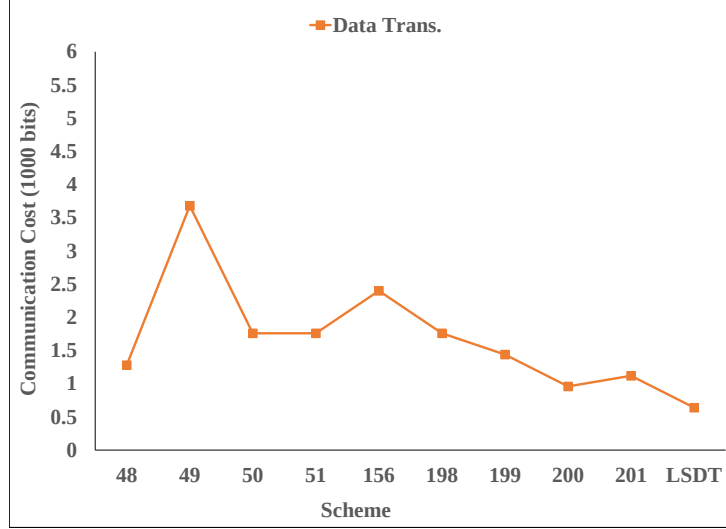


Figure 8.4: Communication Cost - Data Transmission Phase

Notably, the comparison of the total communication costs demonstrates that LSDT outperforms other related schemes as illustrated in Figure 8.5. This achievement can be attributed to the effective combination of symmetric keys and the utilization of the ECMQV asymmetric key agreement protocol. The scheme's ability to minimize overhead and optimize data transmission contributes to its superiority in terms of communication costs. The analysis provided in Section 7.1.1 offers detailed insights into the optimizations employed by the LSDT scheme to achieve these superior results.

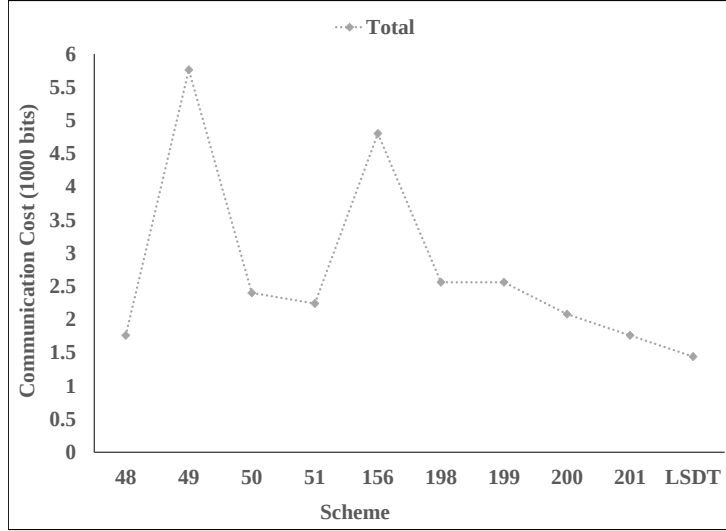


Figure 8.5: Total Communication Cost Comparison

Upon thorough analysis, it is observed that the proposed LSDT scheme exhibits computation costs that are either comparable to or lower than those of [156], [197], [199] during the key generation phase. Simultaneously, it showcases remarkable value when compared to other schemes as shown in Figure 8.6. Furthermore, it is worth noting that the LSDT scheme incurs reduced computation costs during the data transmission phase, which occurs more frequently than the key generation phase as illustrated in Figure 8.7.

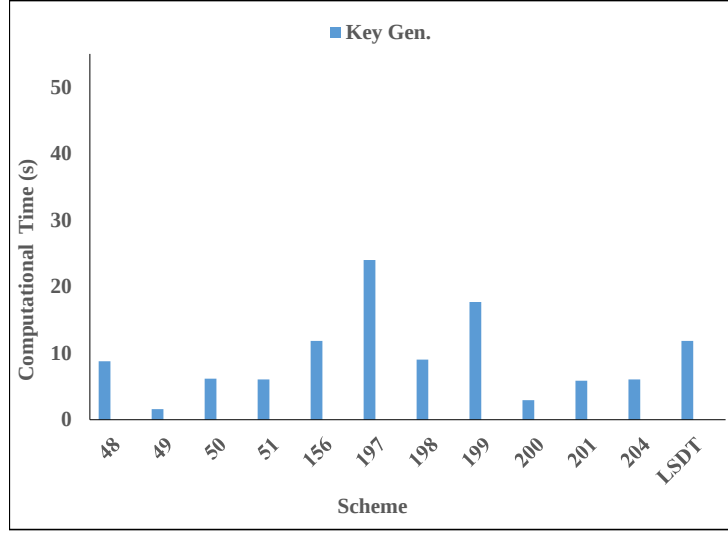


Figure 8.6: Computation Cost - Key Establishment Phase

Significantly, Figure 8.8 showcases the total comparative analysis of the computation costs, revealing the superior performance of the LSDT scheme. By leveraging a well-designed combination of symmetric keys and the ECMQV protocol, the scheme minimizes computational overhead while maintaining a high level of security. This efficient utilization of computational resources makes the LSDT scheme highly suitable for resource-constrained IoT environments. For a comprehensive understanding of the computation costs, please refer to Section 7.1.2, where a detailed breakdown and analysis are provided.



Figure 8.7: Computation Cost - Data Transmission Phase

In conclusion, the results of the comparison demonstrate that the LSDT scheme excels in terms of both communication and computation costs when compared to other related security schemes. This observation serves as clear evidence of the effectiveness and efficiency of the proposed approach compared to existing security schemes. The results of the comparison show that LSDT is secure, lightweight, and efficient compared to other related schemes. Therefore, LSDT is suitable for use in IoT environments with limited resources.

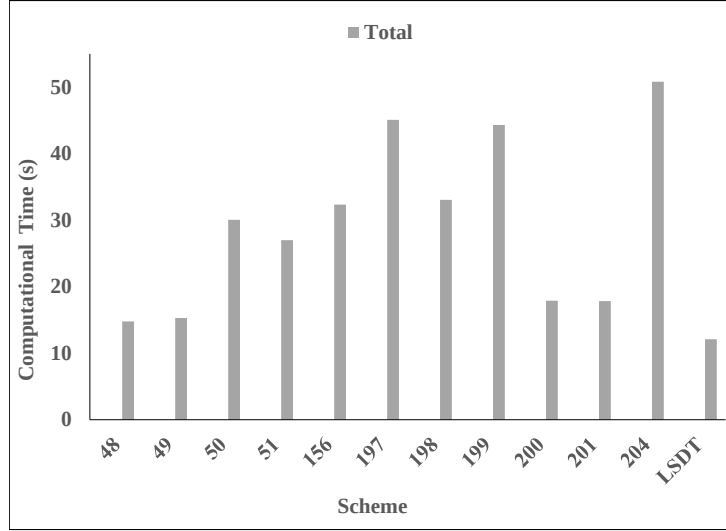


Figure 8.8: Total Computation Cost Comparison

Furthermore, we analyze the communication and computation costs to investigate the cost distribution among LSDT components. Table 8.3 shows the comparison of the communication cost between LSDT and [48], which has similar network architecture to LSDT scheme. It is observed that during key generation phase, LSDT has the same CH communication cost while it has small increased communication cost on CM and BS as compared to [48]. With this small additional cost, LSDT provides data integrity, mutual authentication, and resistance to MITM attack, which are not provided by [48]. It is also noticed that LSDT has less communication cost than [48] during the data transmission phase, which is executed more frequently compared to key generation as shown in Figure 8.9.

Table 8.3: Comparison of Communication Cost

Scheme	Key Gen.			Data Trans.			Total		
	<i>CM</i>	<i>CH</i>	<i>BS</i>	<i>CM</i>	<i>CH</i>	<i>BS</i>	<i>CM</i>	<i>CH</i>	<i>BS</i>
[48]	-	480	-	480	800	-	480	1280	-
LSDT	160	480	160	320	320	-	480	800	160

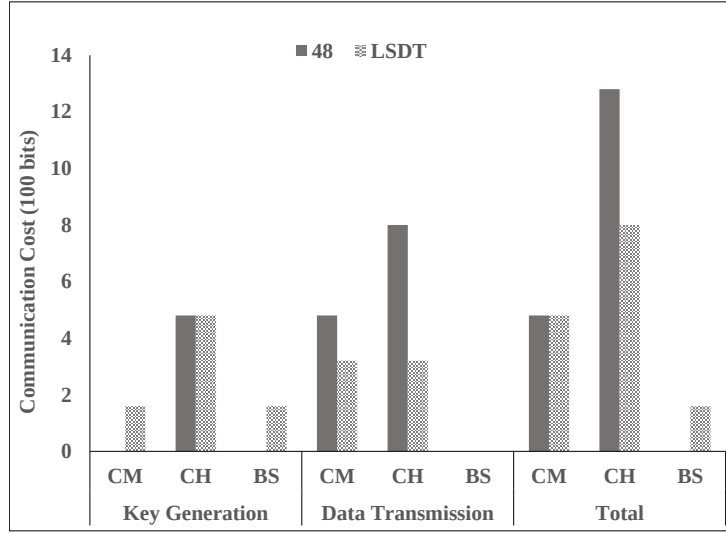


Figure 8.9: Comparison of Communication Cost

The comparison of the computation cost between LSDT and [48] is shown in Table 8.4. It is observed that during the key generation phase, LSDT has less *BS* computation cost than [48] while LSDT has little increased computation cost on *CM* and *CH*. With this little increase, LSDT provides data integrity, mutual authentication, and resistance to MITM attack that are not provided by [48]. In addition, it can be noted that LSDT has less computation cost than [48] during data transmission phase which is carried out more frequently than other phases

as illustrated in Figure 8.10.

Table 8.4: Comparison of Computation Cost

Scheme	Key Gen.			Data Trans.			Total		
	CM	CH	BS	CM	CH	BS	CM	CH	BS
[48]	-	$2T_m$	T_m	T_m	T_h	$T_m + T_h$	T_m	$2T_m + T_h$	$2T_m + T_h$
LSDT	$2T_m$	$2T_m + T_h$	T_h	T_h	$2T_h$	T_h	$2T_m + T_h$	$2T_m + 3T_h$	$2T_h$

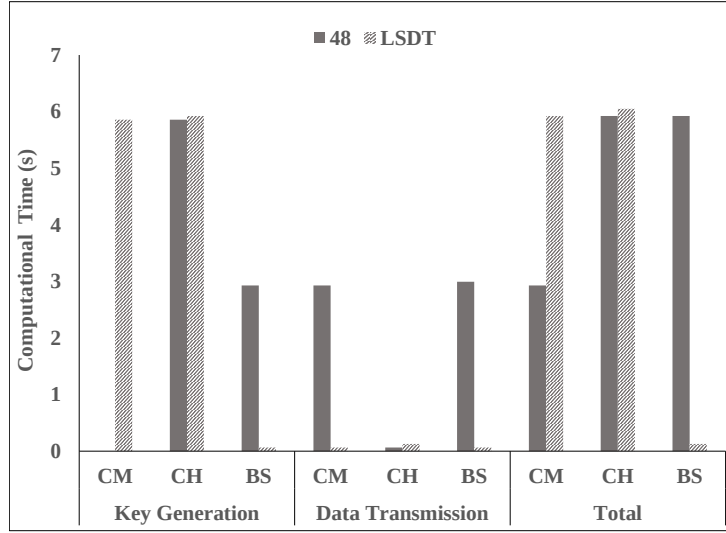


Figure 8.10: Comparison of Computation Cost

To corroborate our analytical findings, our analysis focuses on investigating the distribution of costs associated with communication and computation for multiple data transmissions. For our experimental purposes, three clusters are organized, with each cluster consisting of one cluster head and three cluster members as shown in Figure 8.2. Table 8.5 presents a comparison of the communication costs between the LSDT scheme and [48]. Notably, the LSDT scheme demonstrates

significantly lower communication costs than [48] in both the key generation and data transmission phases. This performance advantage over [48] is depicted in Figure 8.11.

Table 8.5: Comparative Communication Cost Analysis

Scheme	Key Gen.	Data Trans.	Total
[48]	1600	4480	6080
LSDT	2880	2560	5440

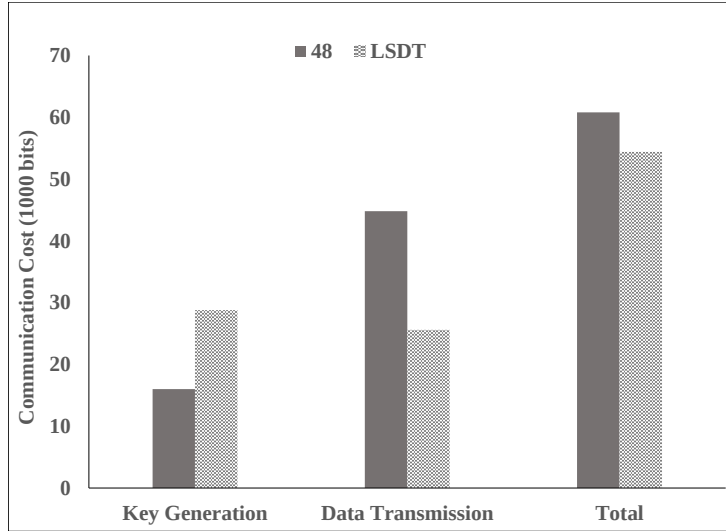


Figure 8.11: Comparative Communication Cost Analysis

In addition, Table 8.6 showcases the comparison of computation costs between LSDT and [48]. It is worth noting that LSDT incurs higher computation costs during the key generation phase compared to [48]. However, this additional computational overhead of LSDT enables the provision of data integrity, mutual authentication, and resistance to MITM attacks, which are not offered by [48]. Fur-

thermore, it is noteworthy that LSDT exhibits considerably lower computation costs than [48] during the data transmission phase which is the most frequently occurring phase, as illustrated in Figure 8.12.

Table 8.6: Comparative Computation Cost Analysis

Scheme	Key Gen.	Data Trans.	Total
[48]	$6T_m + 4T_m$	$8T_m + 4T_h$	$18T_m + 4T_h$
LSDT	$24T_m + 4T_h$	$12T_h + 4T_h$	$24T_m + 20T_h$

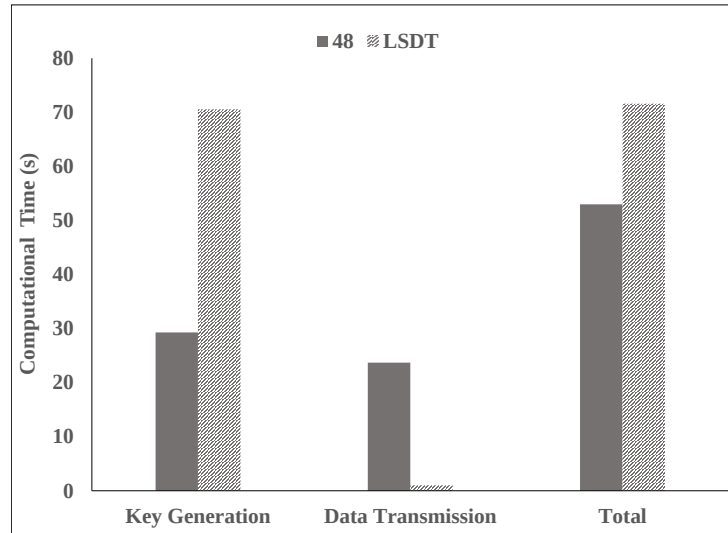


Figure 8.12: Comparative Computation Cost Analysis

CHAPTER 9

CONCLUSION AND FUTURE WORK

The Internet of Things (IoT) has become an integral part of our daily lives, permeating various fields ranging from wearable devices in telehealth to large-scale autonomous systems in transportation and agriculture, to name a few. The current trend in IoT security is towards providing lightweight and efficient schemes to secure data transmission. Ensuring security of IoT applications is a challenge due to the incapability of considerable proportion of resource-limited IoT devices to implement proper security schemes. This chapter concludes the work of this thesis and sketches out outlook on potential future work. In addition, this chapter lists the outcomes from this thesis.

9.1 Conclusions

9.1.1 IoT Data Security Survey

IoT has introduced a revolutionary paradigm that unleashes multitude of functionalities and possibilities. Currently, IoT and its extensive range of applications have become the focal point of numerous research endeavors. While the inherent characteristics of IoT make it highly adaptable to real-world scenarios, they also expose it to potential cybersecurity risks. IoT has experienced an unprecedented number of cyberattacks, which not only affect businesses but also pose potential risks to human lives. Therefore, it is crucial to establish an all-encompassing categorization of challenges in the realm of IoT and identify the existing countermeasures at our disposal. In this thesis, an extensive study on data security attacks, challenges and countermeasures associated with the three data states is conducted. A taxonomy to classify IoT security attacks is presented in chapter 2. The taxonomy is based on different aspects, namely architecture, location and access level of the attacks, data damage level, and objects of vulnerability. In addition, a taxonomy for IoT countermeasures based on the three data states is highlighted in chapter 3. We also discuss future directions towards IoT data security. This survey provides a valuable resource for researchers in both industry and academia who are dedicated to developing secure IoT systems, enabling them to gain insights and make informed decisions in their efforts to enhance the security of IoT systems. The following remarks provide a concise summary encapsulating the key findings and conclusions derived from this comprehensive survey:

- Ensuring the security of data at rest holds utmost significance for two compelling reasons. Firstly, the inherent limitations of IoT devices, including constrained memory capacity, computational power, and bandwidth, pose obstacles to the direct application of conventional security mechanisms, rendering their implementation without adjustments exceedingly challenging. Secondly, the absence of well-defined security guidelines and effective mitigation techniques for safeguarding data at rest in IoT environments further compounds the issue.
- Data in transit is highly susceptible to security attacks, making its protection a critical concern. Particularly in enterprise systems, where confidential data is often transmitted between IoT devices in plaintext, the need to safeguard data from unauthorized access and potential threats becomes paramount. Moreover, the diverse array of communication channels in IoT environments introduces the challenge of employing multiple communication protocols and techniques, which can impede the implementation of certain security solutions.
- It is crucial to acknowledge that no single countermeasure can afford complete protection against cyber attacks. Therefore, adopting a multi-layered approach to security, regularly updating security measures, and staying informed about emerging threats are fundamental practices to mitigate risks effectively.

9.1.2 Lightweight Secure Data Transmission Scheme for IoT Environments

The use of the IoT is expanding at an unprecedented scale in many critical applications due to the ability to interconnect and utilize a plethora of wide-range of devices. However, ensuring secure data transmission in IoT environments remains a significant challenge due to numerous limitations of IoT devices. Traditional cryptographic techniques are no longer appropriate for IoT environments. As such, designing a lightweight secure data transmission scheme is becoming essential. In this thesis, the main contribution is introducing LSDT, a lightweight security scheme that addresses this challenge by providing confidentiality, integrity, and end-to-end authentication through an effective combination of symmetric keys and ECMQV in a scalable scheme. We demonstrate the effectiveness and efficiency of the proposed LSDT scheme through simulation experiments and comparative analysis with related security schemes. The results indicate that the LSDT scheme exhibits robust resistance to various security attacks while maintaining the utmost confidentiality, integrity, and end-to-end mutual authentication. Moreover, the results of the comparison demonstrate that the LSDT scheme excels in terms of both communication and computation costs when compared to other related security schemes. The scheme's lightweight nature, combined with its robust security measures, positions it as a suitable choice for IoT environments with limited resources. The findings presented in this thesis contribute to the advancement of secure data transmission in IoT applications and highlight the potential of the

LSDT scheme in addressing the challenges faced in such environments.

9.2 Limitations

While the proposed LSDT scheme appears promising, there are some limitations to consider. Firstly, the evaluation of the scheme's performance is conducted within a simulation environment, which may not accurately reflect real-world conditions. Additionally, the scheme's scalability requires further investigation to determine its feasibility. Scalability is a crucial aspect to consider in IoT environments, as the number of connected devices and the volume of transmitted data continue to grow rapidly. It is essential to evaluate whether the LSDT scheme can accommodate a large-scale deployment without compromising its performance and efficiency. Furthermore, it is important to study the potential impact of network latency and packet loss on the effectiveness of the LSDT scheme, as these factors hold significant importance in IoT environments. By conducting a comprehensive study on how these issues can affect the scheme's performance, we can better understand the practical implications and optimize the scheme accordingly. This analysis should encompass scenarios where network connectivity is unstable or prone to disruptions, as well as situations where packets may be lost or delayed during transmission. Such insights will allow us to develop robust strategies to mitigate the impact of these challenges and ensure reliable and secure data transmission in IoT environments.

9.3 Future Directions

- Enhancing the trustworthiness of IoT environments by integrating trust-establishing techniques with the proposed LSDT scheme. By empowering nodes to determine the level of trust to place in each other, we strive to promote dependable and stable service-driven IoT systems. Establishing trust in IoT networks is crucial for fostering secure and reliable communication, ultimately leading to enhanced IoT performance and user experience.
- It is worth considering the application of innovative techniques such as blockchain to enhance the security and authentication aspects of the scheme. Blockchain technology, known for its decentralized and tamper-resistant nature, can provide a robust solution for establishing end-to-end authentication across all components of the scheme. By leveraging blockchain, the scheme can ensure trust, integrity, and transparency in data transmission, mitigating the risks of unauthorized access and data tampering. Furthermore, a thorough comparative analysis against other relevant schemes can be conducted to provide valuable insights into the efficiency and practicality of the scheme for data transmission in the IoT environments.
- The proposed scheme could be expanded to include heterogeneous WSNs with numerous base stations. The new scheme can utilize a hierarchical framework within each cluster to facilitate data transmission, aiming to achieve a more uniform distribution of energy consumption among nodes. By implementing this hierarchical approach, the scheme can effectively de-

crease energy consumption, leading to an improved network lifetime. Additionally, the new scheme's performance and scalability requires further investigation to determine whether the scheme can accommodate a large-scale deployment without compromising its performance and efficiency, as these factors are crucial aspects to consider in IoT environments.

9.4 Thesis Outcomes

This section presents the publications that have been emanated from this thesis.

- **Accepted Journal Article:** Al-Hejri, I., Azzedin, F., Almuhammadi, S., and Syed, N. F. "Enabling Efficient Data Transmission in Wireless Sensor Networks Based IoT Applications." *Computers, Materials & Continua (CMC)*. 2023. To Appear.
- **Accepted Journal Article:** Al-Hejri, I., Azzedin, F., Almuhammadi, S., and Eltoweissy, M. "Lightweight Secure and Scalable Scheme for Data Transmission in the Internet of Things." *Arabian Journal for Science and Engineering (AJSE)*. 2023. To Appear.
- **US Patent Assigned Green Light:** Ibraheem Al-Hejri and Farag Azzedin. "Secure Data Transmission Scheme for Internet of Things." KFUPM Reference # 2023-036-TM.
- **Published Conference Paper:** Azzedin, F., & Al-Hejri, I. "A Layered Taxonomy of Internet of Things Attacks". In *Proceedings of the 6th Inter-*

national Conference on Future Networks & Distributed Systems. December 15–16, 2022, Tashkent, Uzbekistan. (pp. 631-636).

- **Accepted Conference Paper:** Azzedin, F., & Al-Hejri, I. “Internet-of-Things Data Security: Challenges, Countermeasures and Opportunities.” In Proceedings of the 7th International Conference on Future Networks & Distributed Systems. December 21-22, 2023. Dubai, UAE.

REFERENCES

- [1] A. Verri Lucca, G. A. Mariano Sborz, V. R. Q. Leithardt, M. Beko, C. Albenes Zeferino, and W. D. Parreira, “A review of techniques for implementing elliptic curve point multiplication on hardware,” *Journal of Sensor and Actuator Networks*, vol. 10, no. 1, p. 3, 2020.
- [2] Y. A. Qadri, A. Nauman, Y. B. Zikria, A. V. Vasilakos, and S. W. Kim, “The future of healthcare internet of things: a survey of emerging technologies,” *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 1121–1167, 2020.
- [3] M. Hammoudeh, R. Newman, C. Dennett, S. Mount, and O. Aldabbas, “Map as a service: a framework for visualising and maximising information return from multi-modal wireless sensor networks,” *Sensors*, vol. 15, no. 9, pp. 22 970–23 003, 2015.
- [4] F. Azzedin and M. Ghaleb, “Internet-of-things and information fusion: Trust perspective survey,” *Sensors*, vol. 19, no. 8, 2019. [Online]. Available: <https://www.mdpi.com/1424-8220/19/8/1929>

- [5] S. Ahmadi, *5G NR: Architecture, Technology, Implementation, and Operation of 3GPP New Radio Standards*. Academic Press, 2019.
- [6] I. Wallgren, “What is the internet of things and why is it important?” 2021. [Online]. Available: <https://www.zooma.se/en/learn/what-is-internet-of-things-and-why-is-it-important>
- [7] W. H. Hassan *et al.*, “Current research on internet of things (iot) security: A survey,” *Computer networks*, vol. 148, pp. 283–294, 2019.
- [8] J. Beyza and J. M. Yusta, “Characterising the security of power system topologies through a combined assessment of reliability, robustness, and resilience,” *Energy Strategy Reviews*, vol. 43, p. 100944, 2022.
- [9] E. Barka, C. A. Kerrache, H. Benkraouda, K. Shuaib, F. Ahmad, and F. Kurugollu, “Towards a trusted unmanned aerial system using blockchain for the protection of critical infrastructure,” *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 8, p. e3706, 2022.
- [10] H. Imran, M. Salama, C. Turner, and S. Fattah, “Cybersecurity risk management frameworks in the oil and gas sector: A systematic literature review,” in *Future of Information and Communication Conference*. Springer, 2022, pp. 871–894.
- [11] S. H. Al Zaabi and R. Zamri, “Managing security threats through touchless security technologies: An overview of the integration of facial recognition

- technology in the UAE oil and gas industry,” *Sustainability*, vol. 14, no. 22, p. 14915, 2022.
- [12] Z. Yang, B. Barroca, A. Weppe, A. Bony-Dandrieux, K. Laffréchine, N. Dacclin, V. November, K. Omrane, D. Kamissoko, F. Benaben *et al.*, “Indicator-based resilience assessment for critical infrastructures—A review,” *Safety science*, vol. 160, p. 106049, 2023.
- [13] G. Ampratwum, R. Osei-Kyei, and V. W. Tam, “Exploring the concept of public-private partnership in building critical infrastructure resilience against unexpected events: A systematic review,” *International Journal of Critical Infrastructure Protection*, p. 100556, 2022.
- [14] N. A. Jalali and H. Chen, “Federated learning security and privacy-preserving algorithm and experiments research under internet of things critical infrastructure,” *Tsinghua Science and Technology*, vol. 29, no. 2, pp. 400–414, 2023.
- [15] V. KHAUSTOVA, M. R. TIRLEA, L. DANDARA, N. TRUSHKINA, and I. BIRCA, “Development of critical infrastructure from the point of view of information security,” *Univers Strategic*, vol. 53 (1), no. 2023, 2023.
- [16] Q. Abu Al-Haija, A. Al Badawi, and G. R. Bojja, “Boost-Defence for resilient IoT networks: A head-to-toe approach,” *Expert Systems*, vol. 39, no. 10, p. e12934, 2022.

- [17] P. Nayak and G. Swapna, “Security issues in IoT applications using certificateless aggregate signcryption schemes: An overview,” *Internet of Things*, p. 100641, 2022.
- [18] F. Stolz, M. Fyrbiak, P. Sasdrich, and T. Güneysu, “Recommendation for a holistic secure embedded isa extension,” in *International Conference on Applied Cryptography and Network Security*. Springer, 2023, pp. 62–84.
- [19] P. Chandani and M. Sharma, “Secure data transmission using cryptography for internet of things and sensor networks applications,” in *2023 2nd International Conference for Innovation in Technology (INOCON)*. IEEE, 2023, pp. 1–7.
- [20] F. Varghese and P. Sasikala, “A detailed review based on secure data transmission using cryptography and steganography,” *Wireless Personal Communications*, pp. 1–28, 2023.
- [21] L. Tawalbeh, F. Muheidat, M. Tawalbeh, M. Quwaider *et al.*, “IoT privacy and security: Challenges and solutions,” *Applied Sciences*, vol. 10, no. 12, p. 4102, 2020.
- [22] A. Karale, “The challenges of IoT addressing security, ethics, privacy, and laws,” *Internet of Things*, vol. 15, p. 100420, 2021.
- [23] G. S. Gaba, G. Kumar, T.-H. Kim, H. Monga, and P. Kumar, “Secure device-to-device communications for 5G enabled internet of things applications,” *Computer Communications*, vol. 169, pp. 114–128, 2021.

- [24] J. Y. Lee and J. Lee, “Current research trends in IoT security: a systematic mapping study,” *Mobile Information Systems*, vol. 2021, pp. 8 847 099:1–8 847 099:25, 2021.
- [25] L. Atzori, A. Iera, and G. Morabito, “Understanding the internet of things: definition, potentials, and societal role of a fast evolving paradigm,” *Ad Hoc Networks*, vol. 56, pp. 122–140, 2017.
- [26] B. Gupta and M. Quamara, “An overview of internet of things (iot): Architectural aspects, challenges, and protocols,” *Concurrency and Computation: Practice and Experience*, vol. 32, no. 21, p. e4946, 2020.
- [27] A. Alwarafy, K. A. Al-Thelaya, M. Abdallah, J. Schneider, and M. Hamdi, “A survey on security and privacy issues in edge computing-assisted internet of things,” *IEEE Internet of Things Journal*, 2020.
- [28] J. Ni, K. Zhang, X. Lin, and X. S. Shen, “Securing fog computing for internet of things applications: Challenges and solutions,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 601–628, 2017.
- [29] K. M. Sadique, R. Rahmani, and P. Johannesson, “Towards security on internet of things: applications and challenges in technology,” *Procedia Computer Science*, vol. 141, pp. 199–206, 2018.
- [30] V. Adat and B. Gupta, “Security in internet of things: issues, challenges, taxonomy, and architecture,” *Telecommunication Systems*, vol. 67, no. 3, pp. 423–441, 2018.

- [31] S. Dhawan and R. Gupta, “Analysis of various data security techniques of steganography: A survey,” *Information Security Journal: A Global Perspective*, vol. 30, no. 2, pp. 63–87, 2021.
- [32] F. Azzedin and T. Alhazmi, “Secure data distribution architecture in IoT using MQTT,” *Applied Sciences*, vol. 13, no. 4, p. 2515, 2023.
- [33] P. R. Kumar, P. H. Raj, and P. Jelciana, “Exploring data security issues and solutions in cloud computing,” *Procedia Computer Science*, vol. 125, pp. 691–697, 2018.
- [34] K. Wanjale, M. Mangla, and P. Marathe, “Security of sensitive data in cloud computing,” *Machine Learning Approach for Cloud Data Analytics in IoT*, pp. 99–118, 2021.
- [35] F. Azzedin, H. Suwad, and M. M. Rahman, “An Asset-Based approach to mitigate Zero-Day ransomware attacks,” *CMC-COMPUTERS MATERIALS & CONTINUA*, vol. 73, no. 2, pp. 3003–3020, 2022.
- [36] F. Li, C. Cui, D. Wang, Z. Liu, N. Elmrabit, Y. Wang, and H. Zhou, “Privacy-aware secure anonymous communication protocol in CPSS cloud computing,” *IEEE Access*, vol. 8, pp. 62 660–62 669, 2020.
- [37] A. Khalique, I. Hussain, M. A. Alam, and T. A. Khan, “Scalable security based on data classification using generalized rsa in cloud storage,” in *Proceedings of ICETIT 2019*. Springer, 2020, pp. 973–983.

- [38] F. Azzedin, H. Suwad, and Z. Alyafeai, “Countermeasureing zero day attacks: Asset-Based approach,” in *2017 International Conference on High Performance Computing & Simulation (HPCS)*, 2017, pp. 854–857.
- [39] L. Li, S. Li, H. Peng, and J. Bi, “An efficient secure data transmission and node authentication scheme for wireless sensing networks,” *Journal of Systems Architecture*, vol. 133, p. 102760, 2022.
- [40] U. Panahi and C. Bayılmış, “Enabling secure data transmission for wireless sensor networks based (IoT) applications,” *Ain Shams Engineering Journal*, vol. 14, no. 2, p. 101866, 2023.
- [41] S. Hariprasad, T. Deepa, and N. Bharathiraja, “HLWEA-IoT: Hybrid lightweight encryption algorithm based secure data transmission in IoT-MQTT networks,” in *Intelligent Cyber Physical Systems and Internet of Things: ICoICI 2022*. Springer, 2023, pp. 813–823.
- [42] N. Mahlake, T. E. Mathonsi, D. Du Plessis, and T. Muchenje, “A lightweight encryption algorithm to enhance wireless sensor network security on the internet of things,” *Journal of Communications*, vol. 18, pp. 47–57, 2023.
- [43] R. F. Olanrewaju, T. Islam, O. O. Khalifa, and F. E. Fajingbesi, “Data in transit validation for cloud computing using cloud-based algorithm detection of injected objects,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 10, no. 1, pp. 348–353, 2018.

- [44] T. Alam, “Efficient and secure data transmission approach in cloud-manet-
iot integrated framework,” *Tanweer Alam. Efficient and Secure Data Trans-
mission Approach in Cloud-MANET-IoT Integrated Framework. Journal of
Telecommunication, Electronic and Computer Engineering*, vol. 12, no. 1,
2020.
- [45] R. AL MOGBIL, M. AL ASQAH, and S. EL KHEDIRI, “Iot: Security chal-
lenges and issues of smart homes/cities,” in *2020 International Conference
on Computing and Information Technology (ICCIT-1441)*. IEEE, 2020, pp.
1–6.
- [46] X. Luo, L. Yin, C. Li, C. Wang, F. Fang, C. Zhu, and Z. Tian, “A lightweight
Privacy-Preserving communication protocol for heterogeneous IoT environ-
ment,” *IEEE Access*, vol. 8, pp. 67 192–67 204, 2020.
- [47] H. I. M. Suwad and F. A. M. Azzedin, “Asset-based security systems and
methods,” May 31 2022, uS Patent 11,347,843.
- [48] Y. Harbi, Z. Aliouat, S. Harous, and A. Bentaleb, “Secure data transmission
scheme based on elliptic curve cryptography for internet of things,” in *Inter-
national Symposium on Modelling and Implementation of Complex Systems*.
Springer, 2018, pp. 34–46.
- [49] F. Wu, L. Xu, S. Kumari, and X. Li, “A privacy-preserving and provable
user authentication scheme for wireless sensor networks based on internet of

- things security,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 8, no. 1, pp. 101–116, 2017.
- [50] S. Kumari, M. Karuppiah, A. K. Das, X. Li, F. Wu, and N. Kumar, “A secure authentication scheme based on elliptic curve cryptography for iot and cloud servers,” *The Journal of Supercomputing*, vol. 74, no. 12, pp. 6428–6453, 2018.
- [51] S. Kalra and S. K. Sood, “Secure authentication scheme for iot and cloud servers,” *Pervasive and Mobile Computing*, vol. 24, pp. 210–223, 2015.
- [52] D. Fang, Y. Qian, and R. Q. Hu, “A flexible and efficient authentication and secure data transmission scheme for iot applications,” *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3474–3484, 2020.
- [53] S. Jebri, A. B. Amor, M. Abid, and A. Bouallegue, “Enhanced lightweight algorithm to secure data transmission in iot systems,” *Wireless Personal Communications*, pp. 1–24, 2020.
- [54] S. Baek and Y.-G. Kim, “C4I system security architecture: A perspective on big data lifecycle in a military environment,” *Sustainability*, vol. 13, no. 24, p. 13827, 2021.
- [55] R. Sharma and R. Arya, “Secure transmission technique for data in IoT edge computing infrastructure,” *Complex & Intelligent Systems*, vol. 8, no. 5, pp. 3817–3832, 2022.

- [56] H. Li, “Pseudo-random scalar multiplication based on group isomorphism,” *Journal of Information Security and Applications*, vol. 53, p. 102534, 2020.
- [57] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, “Internet of things security: A survey,” *Journal of Network and Computer Applications*, vol. 88, pp. 10–28, 2017.
- [58] F. Azzedin and I. Alhejri, “A layered taxonomy of internet of things attacks,” in *Proceedings of the 6th International Conference on Future Networks & Distributed Systems*, 2022, pp. 631–636.
- [59] V. K. Rathi, V. Chaudhary, N. K. Rajput, B. Ahuja, A. K. Jaiswal, D. Gupta, M. Elhoseny, and M. Hammoudeh, “A blockchain-enabled multi domain edge computing orchestrator,” *IEEE Internet of Things Magazine*, vol. 3, no. 2, pp. 30–36, 2020.
- [60] M. Frustaci, P. Pace, G. Aloï, and G. Fortino, “Evaluating critical security issues of the iot world: Present and future challenges,” *IEEE Internet of things journal*, vol. 5, no. 4, pp. 2483–2495, 2017.
- [61] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, “A survey on security and privacy issues in internet-of-things,” *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1250–1258, 2017.
- [62] H. HaddadPajouh, A. Dehghantanha, R. M. Parizi, M. Aledhari, and H. Karimipour, “A survey on internet of things security: Requirements, challenges, and solutions,” *Internet of Things*, p. 100129, 2019.

- [63] M. Jensen, J. Schwenk, N. Gruschka, and L. L. Iacono, “On technical security issues in cloud computing,” in *2009 IEEE International Conference on Cloud Computing*. Ieee, 2009, pp. 109–116.
- [64] K. Chen, S. Zhang, Z. Li, Y. Zhang, Q. Deng, S. Ray, and Y. Jin, “Internet-of-things security and vulnerabilities: Taxonomy, challenges, and practice,” *Journal of Hardware and Systems Security*, vol. 2, no. 2, pp. 97–110, 2018.
- [65] J. Kuusijärvi, R. Savola, P. Savolainen, and A. Evesti, “Mitigating iot security threats with a trusted network element,” in *2016 11th International Conference for Internet Technology and Secured Transactions (ICITST)*. IEEE, 2016, pp. 260–265.
- [66] M. M. Ogonji, G. Okeyo, and J. M. Wafula, “A survey on privacy and security of internet of things,” *Computer Science Review*, vol. 38, p. 100312, 2020.
- [67] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, “A survey on iot security: application areas, security threats, and solution architectures,” *IEEE Access*, vol. 7, pp. 82 721–82 743, 2019.
- [68] N. Y. Kim, S. Rathore, J. H. Ryu, J. H. Park, and J. H. Park, “A survey on cyber physical system security for iot: Issues, challenges, threats, solutions.” *Journal of Information Processing Systems*, vol. 14, no. 6, 2018.

- [69] D. K. Alferidah and N. Jhanjhi, “A review on security and privacy issues and challenges in internet of things,” *International Journal of Computer Science and Network Security IJCSNS*, vol. 20, no. 4, pp. 263–286, 2020.
- [70] M. Burhan, R. A. Rehman, B. Khan, and B.-S. Kim, “Iot elements, layered architectures and security issues: A comprehensive survey,” *Sensors*, vol. 18, no. 9, p. 2796, 2018.
- [71] K. Kimani, V. Oduol, and K. Langat, “Cyber security challenges for iot-based smart grid networks,” *International Journal of Critical Infrastructure Protection*, vol. 25, pp. 36–49, 2019.
- [72] I. Makhdoom, M. Abolhasan, J. Lipman, R. P. Liu, and W. Ni, “Anatomy of threats to the internet of things,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1636–1675, 2018.
- [73] M. Nawir, A. Amir, N. Yaakob, and O. B. Lynn, “Internet of things (iot): Taxonomy of security attacks,” in *2016 3rd International Conference on Electronic Design (ICED)*. IEEE, 2016, pp. 321–326.
- [74] J. Deogirikar and A. Vidhate, “Security attacks in iot: A survey,” in *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*. IEEE, 2017, pp. 32–37.
- [75] H. Mrabet, S. Belguith, A. Alhomoud, and A. Jemai, “A survey of iot security based on a layered architecture of sensing and data analysis,” *Sensors*, vol. 20, no. 13, p. 3625, 2020.

- [76] S. N. Mahapatra, B. K. Singh, and V. Kumar, “A survey on secure transmission in internet of things: Taxonomy, recent techniques, research requirements, and challenges,” *Arabian Journal for Science and Engineering*, vol. 45, no. 8, pp. 6211–6240, 2020.
- [77] C. Pu and S. Hajjar, “Mitigating forwarding misbehaviors in rpl-based low power and lossy networks,” in *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2018, pp. 1–6.
- [78] J. Sengupta, S. Ruj, and S. D. Bit, “A comprehensive survey on attacks, security issues and blockchain solutions for iot and iiot,” *Journal of Network and Computer Applications*, vol. 149, p. 102481, 2020.
- [79] A. Mosenia and N. K. Jha, “A comprehensive study of security of internet-of-things,” *IEEE Transactions on Emerging Topics in Computing*, vol. 5, no. 4, pp. 586–602, 2016.
- [80] A. W. Ahmed, M. M. Ahmed, O. A. Khan, and M. A. Shah, “A comprehensive analysis on the security threats and their countermeasures of iot,” *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 7, pp. 489–501, 2017.
- [81] P. Varga, S. Plosz, G. Soos, and C. Hegedus, “Security threats and issues in automation iot,” in *2017 IEEE 13th International Workshop on Factory Communication Systems (WFCS)*. IEEE, 2017, pp. 1–6.

- [82] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis, and E. K. Markakis, “A survey on the internet of things (iot) forensics: Challenges, approaches and open issues,” *IEEE Communications Surveys & Tutorials*, 2020.
- [83] P. I. R. Grammatikis, P. G. Sarigiannidis, and I. D. Moscholios, “Securing the internet of things: Challenges, threats and solutions,” *Internet of Things*, vol. 5, pp. 41–70, 2019.
- [84] H. A. Abdul-Ghani, D. Konstantas, and M. Mahyoub, “A comprehensive iot attacks survey based on a building-blocked reference model,” *IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 9, no. 3, pp. 355–373, 2018.
- [85] M. H. Ibrahim, “Octopus: An edge-fog mutual authentication scheme.” *IJ Network Security*, vol. 18, no. 6, pp. 1089–1101, 2016.
- [86] N. Park and N. Kang, “Mutual authentication scheme in secure internet of things technology for comfortable lifestyle,” *Sensors*, vol. 16, no. 1, p. 20, 2016.
- [87] B. Cheng, D. Zhu, S. Zhao, and J. Chen, “Situation-aware iot service coordination using the event-driven soa paradigm,” *IEEE Transactions on Network and Service Management*, vol. 13, no. 2, pp. 349–361, 2016.

- [88] A. K. Sikder, G. Petracca, H. Aksu, T. Jaeger, and A. S. Uluagac, “A survey on sensor-based threats to internet-of-things (iot) devices and applications,” *arXiv preprint arXiv:1802.02041*, 2018.
- [89] S. Vadlamani, B. Eksioglu, H. Medal, and A. Nandi, “Jamming attacks on wireless networks: A taxonomic survey,” *International Journal of Production Economics*, vol. 172, pp. 76–94, 2016.
- [90] Y. Guan and X. Ge, “Distributed attack detection and secure estimation of networked cyber-physical systems against false data injection attacks and jamming attacks,” *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 1, pp. 48–59, 2017.
- [91] H. Xie, Z. Yan, Z. Yao, and M. Atiquzzaman, “Data collection for security measurement in wireless sensor networks: a survey,” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2205–2224, 2018.
- [92] O. A. Osanaiye, A. S. Alfa, and G. P. Hancke, “Denial of service defence for resource availability in wireless sensor networks,” *IEEE Access*, vol. 6, pp. 6975–7004, 2018.
- [93] S. A. Kumar, T. Vealey, and H. Srivastava, “Security in internet of things: Challenges, solutions and future directions,” in *2016 49th Hawaii International Conference on System Sciences (HICSS)*. IEEE, 2016, pp. 5772–5781.
- [94] S. Kumar, S. Sahoo, A. Mahapatra, A. K. Swain, and K. K. Mahapatra, “Security enhancements to system on chip devices for iot perception layer,”

- in *2017 IEEE International Symposium on Nanoelectronic and Information Systems (iNIS)*. IEEE, 2017, pp. 151–156.
- [95] J. Deogirikar and A. Vidhate, “Security attacks in iot: A survey,” in *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*. IEEE, 2017, pp. 32–37.
- [96] W. Stallings, *Network security essentials: applications and standards*. Pearson, 2016.
- [97] A. Mayzaud, R. Badonnel, and I. Chrisment, “A taxonomy of attacks in rpl-based internet of things,” *International Journal of Network Security*, vol. 18, no. 3, pp. 459–473, 2016.
- [98] LibreTexts, “Information security,” 2021. [Online]. Available: https://eng.libretexts.org/Courses/Delta_College/Information_Security
- [99] G. Database, “Types of threats,” 2021. [Online]. Available: <https://genesisdatabase.wordpress.com/2010/12/13/types-of-threats-interception-interruption-modification-fabrication/>
- [100] M. M. Ahemd, M. A. Shah, and A. Wahid, “Iot security: A layered approach for attacks & defenses,” in *2017 international conference on Communication Technologies (ComTech)*. IEEE, 2017, pp. 104–110.
- [101] P. Varga, S. Plosz, G. Soos, and C. Hegedus, “Security threats and issues in automation iot,” in *2017 IEEE 13th International Workshop on Factory Communication Systems*. IEEE, 2017, pp. 1–6.

- [102] A. Kayes, R. Kalaria, I. H. Sarker, M. S. Islam, P. A. Watters, A. Ng, M. Hammoudeh, S. Badsha, and I. Kumara, “A survey of context-aware access control mechanisms for cloud and fog networks: Taxonomy and open research issues,” *Sensors*, vol. 20, no. 9, p. 2464, 2020.
- [103] H. A. Abdulghani, N. A. Nijdam, A. Collen, and D. Konstantas, “A study on security and privacy guidelines, countermeasures, threats: Iot data at rest perspective,” *Symmetry*, vol. 11, no. 6, p. 774, 2019.
- [104] F. Azzedin and M. Ghaleb, “Internet-of-things and information fusion: Trust perspective survey,” *Sensors*, vol. 19, no. 8, p. 1929, 2019.
- [105] T. Husein, A. Khalique, M. A. Alam, and B. Alankar, “A dynamic scalable security model for data at rest in fog computing environment,” 2020.
- [106] L. Yang, X. Xu, J. Gao, L. Chen, and C. Feng, “Storing multiple encryption keys to protect data at rest,” Aug. 28 2018, uS Patent 10,061,717.
- [107] S. F. Tan and A. Samsudin, “Recent technologies, security countermeasure and ongoing challenges of industrial internet of things (iiot): A survey,” *Sensors*, vol. 21, no. 19, p. 6647, 2021.
- [108] M. E. Hussain and R. Hussain, “Cloud security as a service using data loss prevention: Challenges and solution,” in *International Conference on Internet of Things and Connected Technologies*. Springer, 2021, pp. 98–106.
- [109] S. F. Bentley, “Data-at-rest (dar) encryption for integrated storage media,” Nov. 12 2019, uS Patent 10,476,846.

- [110] “Protecting the three states of data,” <https://www.sealpath.com/protecting-the-three-states-of-data/>, note = Accessed on 27.10.2020.
- [111] “Data at rest,” https://en.wikipedia.org/wiki/Data_at_rest, note = Accessed on 27.10.2020.
- [112] “Data in use,” https://en.wikipedia.org/wiki/Data_in_use, note = Accessed on 27.10.2020.
- [113] T. Braun, B. C. Fung, F. Iqbal, and B. Shah, “Security and privacy challenges in smart cities,” *Sustainable cities and society*, vol. 39, pp. 499–507, 2018.
- [114] C. Lee, L. Zappaterra, K. Choi, and H.-A. Choi, “Securing smart home: Technologies, security challenges, and security requirements,” in *2014 IEEE Conference on Communications and Network Security*. IEEE, 2014, pp. 67–72.
- [115] M. G. Samaila, M. Neto, D. A. Fernandes, M. M. Freire, and P. R. Inácio, “Security challenges of the internet of things,” in *Beyond the Internet of Things*. Springer, 2017, pp. 53–82.
- [116] W. Zhou, Y. Jia, A. Peng, Y. Zhang, and P. Liu, “The effect of iot new features on security and privacy: New threats, existing solutions, and challenges yet to be solved,” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1606–1616, 2018.

- [117] G. Kulkarni, J. Gambhir, T. Patil, and A. Dongare, “A security aspects in cloud computing,” in *2012 IEEE International Conference on Computer Science and Automation Engineering*. IEEE, 2012, pp. 547–550.
- [118] A. Albugmi, M. O. Alassafi, R. Walters, and G. Wills, “Data security in cloud computing,” in *2016 Fifth International Conference on Future Generation Communication Technologies (FGCT)*. IEEE, 2016, pp. 55–59.
- [119] S. Singh, Y.-S. Jeong, and J. H. Park, “A survey on cloud computing security: Issues, threats, and solutions,” *Journal of Network and Computer Applications*, vol. 75, pp. 200–222, 2016.
- [120] N. Sinha and L. Khreisat, “Cloud computing security, data, and performance issues,” in *2014 23rd Wireless and Optical Communication Conference (WOCC)*. IEEE, 2014, pp. 1–6.
- [121] V. Tchifilionova, “Security and privacy implications of cloud computing—lost in the cloud,” in *International Workshop on Open Problems in Network Security*. Springer, 2010, pp. 149–158.
- [122] S. Achar, “Cloud computing security for multi-cloud service providers: Controls and techniques in our modern threat landscape,” *International Journal of Computer and Systems Engineering*, vol. 16, no. 9, pp. 379–384, 2022.
- [123] L. Kacha and A. Zitouni, “An overview on data security in cloud computing,” in *Proceedings of the Computational Methods in Systems and Software*. Springer, 2017, pp. 250–261.

- [124] K. Hashizume, D. G. Rosado, E. Fernández-Medina, and E. B. Fernandez, “An analysis of security issues for cloud computing,” *Journal of internet services and applications*, vol. 4, no. 1, p. 5, 2013.
- [125] R. Neha and J. Rishabh, “Cloud computing: architecture and concept of virtualization,” *International Journal of Science, Technology & Management*, vol. 4, no. 1, 2015.
- [126] K. Nandakumar, V. Vinod, S. M. Akbar Batcha, D. K. Sharma, M. Elangovan, A. Poonia, S. Mudlappa Basavaraju, S. R. Dogiwal, P. Dadheech, and S. Sengan, “Securing data in transit using data-in-transit defender architecture for cloud communication,” *Soft Computing*, vol. 25, no. 18, pp. 12 343–12 356, 2021.
- [127] X. Liang and Y. Kim, “A survey on security attacks and solutions in the iot network,” in *2021 IEEE 11th annual computing and communication workshop and conference (CCWC)*. IEEE, 2021, pp. 0853–0859.
- [128] M. MOHAMMED and F. ABED, “A symmetric-based framework for securing cloud data at rest,” *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 28, no. 1, pp. 347–361, 2020.
- [129] S. Mistry, “Securing data at rest utilizing cloud provider service agnostic encrypted data transportability,” Jun. 18 2020, uS Patent App. 16/223,622.

- [130] Z. T. Omar and F. S. Abed, “A proposed fully homomorphic for securing cloud banking data at rest,” *UHD Journal of Science and Technology*, vol. 4, no. 1, pp. 87–95, 2020.
- [131] G. Matloob and F. Siddiqui, “Data at rest and it’s security solutions-a survey.” *International Journal of Advanced Research in Computer Science*, vol. 8, no. 5, 2017.
- [132] “Enhanced securing of data at rest,” 2018, uS Patent 10,528,754.
- [133] D. H. Gallancy, E. A. Aronesty, and D. Nemirovsky, “Passwordless security system for data-at-rest,” Sep. 19 2019, uS Patent App. 16/106,564.
- [134] H. Heni, M. B. Abdallah, and F. Gargouri, “Combining fragmentation and encryption to ensure big data at rest security,” in *International Conference on Health Information Science*. Springer, 2017, pp. 177–185.
- [135] S. Kumar, “Securing data at rest using hill cipher and xor based operations,” in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*. IEEE, 2018, pp. 1839–1843.
- [136] P. Matta, B. Pant, and U. K. Tiwari, “Ddita: A naive security model for iot resource security,” in *Smart Innovations in Communication and Computational Sciences*. Springer, 2019, pp. 199–209.
- [137] A. H. Lone and A. Khalique, “Generalized rsa using 2k prime numbers with secure key generation,” *Security and Communication Networks*, vol. 9, no. 17, pp. 4443–4450, 2016.

- [138] J. A. Blackley, J. E. Carlson, M. D. Hanrahan, B. Pattan, and C. S. Alcott, "Secure data at rest," Aug. 20 2020, uS Patent App. 16/277,740.
- [139] O. Mazonka, E. Sarkar, E. Chielle, N. G. Tsoutsos, and M. Maniatakos, "Practical data-in-use protection using binary decision diagrams," *IEEE Access*, vol. 8, pp. 23 847–23 862, 2020.
- [140] W. H. Basri and M. Yusof, "Assessing the data leakage prevention for data-in-use," in *Pacific Asia Conference on Information Systems (PACIS)*. Association For Information Systems, 2017.
- [141] A. Karale, "The challenges of IoT addressing security, ethics, privacy, and laws," *Internet Things*, vol. 15, p. 100420, 2021.
- [142] G. S. Gaba, G. Kumar, T. hoon Kim, H. Monga, and P. Kumar, "Secure device-to-device communications for 5G enabled internet of things applications," *Comput. Commun.*, vol. 169, pp. 114–128, 2021.
- [143] M. Conti, N. Dragoni, and V. Lesyk, "A survey of man in the middle attacks," *IEEE Communications Surveys Tutorials*, vol. 18, no. 3, pp. 2027–2051, 2016.
- [144] H. Lan, X. Zhu, J. Sun, and S. Li, "Traffic data classification to detect man-in-the-middle attacks in industrial control system," in *2019 6th International Conference on Dependable Systems and Their Applications (DSA)*, 2020, pp. 430–434.

- [145] H. Wong and T. Luo, “Man-in-the-middle attacks on mqtt-based iot using bert based adversarial message generation,” *KDD’20*.
- [146] M. Ghouse, M. J. Nene, and C. VembuSelvi, “Data leakage prevention for data in transit using artificial intelligence and encryption techniques,” in *2019 International Conference on Advances in Computing, Communication and Control (ICAC3)*. IEEE, 2019, pp. 1–6.
- [147] F. Li, Z. Zheng, and C. Jin, “Secure and efficient data transmission in the internet of things,” *Telecommunication Systems*, vol. 62, no. 1, pp. 111–122, 2016.
- [148] M. Luo, Y. Wen, and X. Hu, “Practical data transmission scheme for wireless sensor networks in heterogeneous iot environment,” *Wireless Personal Communications*, vol. 109, no. 1, pp. 505–519, 2019.
- [149] M. Kumar, H. Kumar Verma, and G. Sikka, “A secure data transmission protocol for cloud-assisted edge-internet of things environment,” *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 6, p. e3883, 2020.
- [150] C. Wolfe, S. Graham, and P. Simon, “Securing data in transit using tunable two channel communication,” in *International Conference on Cyber Warfare and Security*. Academic Conferences International Limited, 2018, pp. 627–XVI.

- [151] J. Haager, C. Sandwith, J. Terrano, and P. Saripalli, “Systems and methods for security hardening of data in transit and at rest via segmentation, shuffling and multi-key encryption,” Jun. 5 2018, uS Patent 9,990,502.
- [152] A. Varriale, P. Prinetto, A. Carelli, and P. Trotta, “Secube (tm): Data at rest and data in motion protection,” in *Proceedings of the International Conference on Security and Management (SAM)*. The Steering Committee of The World Congress in Computer Science, Computer ..., 2016, p. 138.
- [153] H. Hui, C. Zhou, S. Xu, and F. Lin, “A novel secure data transmission scheme in industrial internet of things,” *China Communications*, vol. 17, no. 1, pp. 73–88, 2020.
- [154] C. Zhang, M. Zhao, Y. Xu, T. Wu, Y. Li, L. Zhu, and H. Wang, “Achieving fuzzy matching data sharing for secure cloud-edge communication,” *China Communications*, vol. 19, no. 7, pp. 257–276, 2022.
- [155] B. Bera, A. K. Das, S. Garg, M. J. Piran, and M. S. Hossain, “Access control protocol for battlefield surveillance in drone-assisted iot environment,” *IEEE Internet of Things Journal*, 2021.
- [156] Y. Harbi, Z. Aliouat, A. Refoufi, S. Harous, and A. Bentaleb, “Enhanced authentication and key management scheme for securing data transmission in the internet of things,” *Ad Hoc Networks*, vol. 94, p. 101948, 2019.

- [157] A. Mehmood, M. M. Umar, and H. Song, “Icmds: Secure inter-cluster multiple-key distribution scheme for wireless sensor networks,” *Ad Hoc Networks*, vol. 55, pp. 97–106, 2017.
- [158] E. Yuan, L. Wang, S. Cheng, N. Ao, and Q. Guo, “A key management scheme based on pairing-free identity based digital signature algorithm for heterogeneous wireless sensor networks,” *Sensors*, vol. 20, no. 6, p. 1543, 2020.
- [159] M. M. Gilanian Sadeghi, “A secure channel to improve energy cost in internet of things,” *Journal of Computer & Robotics*, vol. 12, no. 1, pp. 123–130, 2019.
- [160] H. A. Babaeer and S. A. Al-Ahmadi, “Efficient and secure data transmission and sinkhole detection in a multi-clustering wireless sensor network based on homomorphic encryption and watermarking,” *IEEE Access*, vol. 8, pp. 92 098–92 109, 2020.
- [161] P. R. Raj and D. Anand, “Sink Hole attack detection using two step verification technique in wireless sensor networks,” in *2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*. IEEE, 2021, pp. 1245–1250.
- [162] G. Rathee, M. Balasaraswathi, K. P. Chandran, S. D. Gupta, and C. Boopathi, “A secure iot sensors communication in industry 4.0 using

- blockchain technology,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 1, pp. 533–545, 2021.
- [163] A. A. Abd El-Latif, B. Abd-El-Atty, W. Mazurczyk, C. Fung, and S. E. Venegas-Andraca, “Secure data encryption based on quantum walks for 5g internet of things scenario,” *IEEE Transactions on Network and Service Management*, vol. 17, no. 1, pp. 118–131, 2020.
- [164] B. A. Forouzan and D. Mukhopadhyay, *Cryptography and Network Security (Sie)*. McGraw-Hill Education, 2011.
- [165] K. R. Saraf, V. P. Jagtap, and A. K. Mishra, “Text and image encryption decryption using advanced encryption standard,” *IJETTCS*, vol. 3, no. 3, pp. 118–26, 2014.
- [166] E. Setyaningsih, R. Wardoyo, and A. K. Sari, “Securing color image transmission using compression-encryption model with dynamic key generator and efficient symmetric key distribution,” *Digital Communications and Networks*, vol. 6, no. 4, pp. 486–503, 2020.
- [167] N. Lewis and N. Foukia, “Using trust in key distribution in wireless sensor networks,” in *2007 IEEE Globecom Workshops*. IEEE, 2007, pp. 1–5.
- [168] L. Harn, C. Hsu, and Z. Xia, “Lightweight and flexible key distribution schemes for secure group communications,” *Wireless Networks*, vol. 27, pp. 129–136, 2021.

- [169] D. Wu and C. Zhou, “Fault-tolerant and scalable key management for smart grid,” *IEEE Transactions on Smart Grid*, vol. 2, no. 2, pp. 375–381, 2011.
- [170] S. T. F. Al-Janabi and M. A.-s. Rasheed, “Public-key cryptography enabled kerberos authentication,” in *2011 Developments in E-systems Engineering*. IEEE, 2011, pp. 209–214.
- [171] B. Padma and S. B. Erukala, “Keys distribution among end devices using trust-based blockchainsystem for securing zigbee-enabled IoT networks,” *SSRN Electronic Journal*, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:257628578>
- [172] A. M. Odlyzko, “The rise and fall of knapsack cryptosystems,” *Cryptology and computational number theory*, vol. 42, pp. 75–88, 1990.
- [173] J. Li, Y. Luo, F. Wang, and W. Gao, “Design and implementation of Real-Time image acquisition chip based on Triple-Hybrid encryption system,” *Electronics*, vol. 11, no. 18, p. 2925, 2022.
- [174] P. C. Sethi, N. Sahu, and P. K. Behera, “Group security using ECC,” *International Journal of Information Technology*, pp. 1–9, 2021.
- [175] S. Chandel, W. Cao, Z. Sun, J. Yang, B. Zhang, and T.-Y. Ni, “A multi-dimensional adversary analysis of RSA and ECC in blockchain encryption,” in *Advances in Information and Communication: Proceedings of the 2019 Future of Information and Communication Conference (FICC), Volume 2*. Springer, 2020, pp. 988–1003.

- [176] P. Kumar and A. Kumar Bhatt, “Enhancing multi-tenancy security in the cloud computing using hybrid ECC-based data encryption approach,” *IET Communications*, vol. 14, no. 18, pp. 3212–3222, 2020.
- [177] P. Vijayalakshmi and K. B. Raja, “Performance analysis of RSA and ECC in identity-based authenticated new multiparty key agreement protocol,” in *2012 International Conference on Computing, Communication and Applications*. IEEE, 2012, pp. 1–5.
- [178] W. Cao, H. Chen, J. Feng, L. Fan, and W. Wu, “Lattice-Based fault attacks against ECMQV,” in *International Workshop on Constructive Side-Channel Analysis and Secure Design*. Springer, 2018, pp. 101–116.
- [179] W.-B. Hsieh and J.-S. Leu, “Implementing a secure VoIP communication over SIP-based networks,” *Wireless Networks*, vol. 24, no. 8, pp. 2915–2926, 2018.
- [180] S. Sciancalepore, A. Caposelle, G. Piro, G. Boggia, and G. Bianchi, “Key management protocol with implicit certificates for IoT systems,” in *Proceedings of the 2015 Workshop on IoT challenges in Mobile and Industrial Systems*, 2015, pp. 37–42.
- [181] U. Banerjee, C. Juvekar, A. Wright, A. P. Chandrakasan *et al.*, “An energy-efficient reconfigurable DTLS cryptographic engine for End-to-End security in IoT applications,” in *2018 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2018, pp. 42–44.

- [182] W. Lattin, “Efficient and authenticated key agreement,” *Reprinted from Military Embedded Systems*, 2006.
- [183] S. C. Seo and D. Kwon, “Highly efficient SCA-Resistant binary field multiplication on 8-Bit AVR microcontrollers,” *Applied Sciences*, vol. 10, no. 8, p. 2821, 2020.
- [184] N. Fayed, E. Daydamoni, and A. Atwan, “Efficient combined security system for wireless sensor network,” *Egyptian Informatics Journal*, vol. 13, no. 3, pp. 185–190, 2012.
- [185] N. N. Anandakumar, M. P. L. Das, S. K. Sanadhya, and M. S. Hashmi, “Reconfigurable hardware architecture for authenticated key agreement protocol over binary edwards curve,” *ACM Transactions on Reconfigurable Technology and Systems (TRETs)*, vol. 11, pp. 1 – 19, 2018.
- [186] G. Kambourakis, C. Kolias, D. Geneiatakis, G. Karopoulos, G. M. Makrakis, and I. Kounelis, “A State-of-the-Art review on the security of mainstream IoT wireless pan protocol stacks,” *Symmetry*, vol. 12, no. 4, p. 579, 2020.
- [187] Z. Chen, A. Wu, Y. Li, Q. Xing, and S. Geng, “Blockchain-enabled public key encryption with multi-keyword search in cloud computing,” *Security and Communication Networks*, vol. 2021, pp. 1–11, 2021.
- [188] A. Braeken, “Public key versus symmetric key cryptography in client–server authentication protocols,” *International Journal of Information Security*, vol. 21, no. 1, pp. 103–114, 2022.

- [189] J. Höglund, S. Lindemer, M. Furuheid, and S. Raza, “PKI4IoT: towards public key infrastructure for the internet of things,” *Computers & Security*, vol. 89, p. 101658, 2020.
- [190] M. Gupta, S. Tanwar, T. K. Bhatia, S. Badotra, and Y.-C. Hu, “A comparative study on blockchain-based distributed public key infrastructure for IoT applications,” *Multimedia Tools and Applications*, pp. 1–26, 2023.
- [191] B. Chen, L. Wu, N. Kumar, K.-K. R. Choo, and D. He, “Lightweight searchable public-key encryption with forward privacy over IIoT outsourced data,” *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 4, pp. 1753–1764, 2019.
- [192] P. Chinnnasamy, S. Padmavathi, R. Swathy, and S. Rakesh, “Efficient data security using hybrid cryptography on cloud computing,” in *Inventive Communication and Computational Technologies: Proceedings of ICICCT 2020*. Springer, 2021, pp. 537–547.
- [193] Q. Zhang, “An overview and analysis of hybrid encryption: the combination of symmetric encryption and asymmetric encryption,” in *2021 2nd international conference on computing and data science (CDS)*. IEEE, 2021, pp. 616–622.
- [194] L. Jiang, R. Tan, X. Lou, and G. Lin, “On lightweight privacy-preserving collaborative learning for internet of things by independent random projec-

- tions,” *ACM Transactions on Internet of Things*, vol. 2, no. 2, pp. 1–32, 2021.
- [195] M. Shen, G. Cheng, L. Zhu, X. Du, and J. Hu, “Content-based multi-source encrypted image retrieval in clouds with privacy preservation,” *Future Generation Computer Systems*, vol. 109, pp. 621–632, 2020.
- [196] L. Jiang, R. Tan, X. Lou, and G. Lin, “On lightweight privacy-preserving collaborative learning for internet-of-things objects,” in *Proceedings of the international conference on internet of things design and implementation*, 2019, pp. 70–81.
- [197] T. K. Dang, C. D. Pham, and T. L. Nguyen, “A pragmatic elliptic curve cryptography-based extension for energy-efficient device-to-device communications in smart cities,” *Sustainable Cities and Society*, vol. 56, p. 102097, 2020.
- [198] P. K. Panda and S. Chattopadhyay, “A secure mutual authentication protocol for iot environment,” *Journal of Reliable Intelligent Environments*, vol. 6, pp. 79–94, 2020.
- [199] K. Sowjanya, M. Dasgupta, and S. Ray, “An elliptic curve cryptography based enhanced anonymous authentication protocol for wearable health monitoring systems,” *International Journal of Information Security*, vol. 19, pp. 129–146, 2020.

- [200] X. Li, J. Peng, S. Kumari, F. Wu, M. Karuppiah, and K.-K. R. Choo, “An enhanced 1-round authentication protocol for wireless body area networks with user anonymity,” *Computers & Electrical Engineering*, vol. 61, pp. 238–249, 2017.
- [201] M. L. Das, P. Kumar, and A. Martin, “Secure and privacy-preserving rfid authentication scheme for internet of things applications,” *Wireless Personal Communications*, vol. 110, pp. 339–353, 2020.
- [202] C.-C. Chang, H.-L. Wu, and C.-Y. Sun, “Notes on “secure authentication scheme for iot and cloud servers”,” *Pervasive and Mobile Computing*, vol. 38, pp. 275–278, 2017.
- [203] K.-H. Wang, C.-M. Chen, W. Fang, and T.-Y. Wu, “A secure authentication scheme for internet of things,” *Pervasive and Mobile Computing*, vol. 42, pp. 15–26, 2017.
- [204] M. Ataei Nezhad, H. Barati, and A. Barati, “An Authentication-Based secure data aggregation method in internet of things,” *Journal of Grid Computing*, vol. 20, no. 3, p. 29, 2022.
- [205] F. Afroz and R. Braun, “Empirical analysis of extended QX-MAC for IOT-Based WSNS,” *Electronics*, vol. 11, no. 16, p. 2543, 2022.
- [206] A. Seyyedabbasi, F. Kiani, T. Allahviranloo, U. Fernandez-Gamiz, and S. Noeiaghdam, “Optimal data transmission and pathfinding for WSN and

- decentralized iot systems using I-GWO and Ex-GWO algorithms,” *Alexandria Engineering Journal*, vol. 63, pp. 339–357, 2023.
- [207] S. Wu, X. Min, and J. Li, “Optimal data transmission for WSNs with Data-Location integration,” *Symmetry*, vol. 13, no. 8, p. 1499, 2021.
- [208] K. Vijayalakshmi and P. Anandan, “Global levy flight of cuckoo search with particle swarm optimization for effective cluster head selection in wireless sensor network,” *Intelligent Automation and Soft Computing*, vol. 26, no. 2, pp. 303–311, 2020.
- [209] Y. Miao, G. Wu, C. Liu, M. S. Hossain, and G. Muhammad, “Green cognitive body sensor network: architecture, energy harvesting, and smart clothing-based applications,” *IEEE Sensors Journal*, vol. 19, no. 19, pp. 8371–8378, 2018.
- [210] T. Palanisamy, D. Alghazzawi, S. Bhatia, A. A. Malibari, P. Dadheech, and S. Sengan, “Improved energy based multi-sensor object detection in wireless sensor networks,” *Intelligent Automation & Soft Computing*, vol. 33, no. 1, pp. 227–244, 2022.
- [211] A. S. Abdalkafor and S. A. Aliesawi, “Efficient data aggregation strategy in wireless sensor networks: Challenges and significant applications,” in *Proceedings of International Conference on Computing and Communication Networks*. Springer, 2022, pp. 131–139.

- [212] M. Elhoseny and A. E. Hassanien, “Secure data transmission in WSN: An overview,” *Dynamic wireless sensor networks*, pp. 115–143, 2019.
- [213] G. Halidoddi and R. Pandu, “Secured data transmission using multi-objective trust based Bat optimization algorithm and enhanced homomorphic cryptosystem for WSN,” *International Journal of Intelligent Engineering and Systems*, vol. 15, no. 1, pp. 214–224, 2022.
- [214] V. Sivasankarareddy, G. Sundari, C. Rami Reddy, F. Aymen, and E. C. Bortoni, “Grid-Based routing model for energy efficient and secure data transmission in WSN for smart building applications,” *Applied Sciences*, vol. 11, no. 22, p. 10517, 2021.
- [215] A. Johnson, J. Molloy, J. Yunes, J. Puthuparampil, and A. Elleithy, “Security in wireless sensors networks,” 2019, pp. 1–3.
- [216] J. Großschädl, A. Szekely, and S. Tillich, “The energy cost of cryptographic key establishment in wireless sensor networks,” in *Proceedings of the 2nd ACM symposium on Information, computer and communications security*, 2007, pp. 380–382.
- [217] H. Krawczyk, “HMQV: A high-performance secure diffie-hellman protocol,” in *Annual international cryptology conference*. Springer, 2005, pp. 546–566.
- [218] B. Bera, A. K. Das, S. Garg, M. J. Piran, and M. S. Hossain, “Access control protocol for battlefield surveillance in Drone-Assisted IoT environment,” *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2708–2721, 2022.

- [219] A. K. Maurya, A. K. Das, S. S. Jamal, and D. Giri, “Secure user authentication mechanism for iot-enabled wireless sensor networks based on multiple bloom filters,” *Journal of Systems Architecture*, vol. 120, p. 102296, 2021.
- [220] N. Zhang, Z. Chen, and G. Xiao, “Efficient elliptic curve scalar multiplication algorithms resistant to power analysis,” *Information Sciences*, vol. 177, no. 10, pp. 2119–2129, 2007.
- [221] Y. Sakai and K. Sakurai, “Algorithms for efficient simultaneous elliptic scalar multiplication with reduced joint hamming weight representation of scalars,” in *International Conference on Information Security*. Springer, 2002, pp. 484–499.
- [222] A. Liu and P. Ning, “TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks,” in *2008 International Conference on Information Processing in Sensor Networks (ipsn 2008)*. IEEE, 2008, pp. 245–256.
- [223] S. Di Matteo, L. Baldanzi, L. Crocetti, P. Nannipieri, L. Fanucci, and S. Saponara, “Secure elliptic curve Crypto-Processor for Real-Time IoT applications,” *Energies*, vol. 14, no. 15, p. 4676, 2021.
- [224] E. Gyamfi, J. A. Ansere, and L. Xu, “ECC based lightweight cybersecurity solution for IoT networks utilising multi-access mobile edge computing,” in *2019 Fourth international conference on fog and mobile edge computing (FMEC)*. IEEE, 2019, pp. 149–154.

- [225] P. S. Munoz, N. Tran, B. Craig, B. Dezfouli, and Y. Liu, “Analyzing the resource utilization of AES encryption on IoT devices,” in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2018, pp. 1200–1207.
- [226] P. P. Jayaraman, X. Yang, A. Yavari, D. Georgakopoulos, and X. Yi, “Privacy preserving internet of things: From privacy techniques to a blueprint architecture and efficient implementation,” *Future Generation Computer Systems*, vol. 76, pp. 540–549, 2017.
- [227] M. Rana, Q. Mamun, and R. Islam, “Lightweight cryptography in IoT networks: A survey,” *Future Generation Computer Systems*, vol. 129, pp. 77–89, 2022.
- [228] K. T. Nguyen, “Lightweight security protocols for IP-based wireless sensor networks and the internet of things,” Ph.D. dissertation, Evry, Institut national des télécommunications, 2016.
- [229] S. S. Hameedi and O. Bayat, “Improving IoT data security and integrity using lightweight blockchain dynamic table,” *Applied Sciences*, vol. 12, no. 18, p. 9377, 2022.
- [230] U. Panahi and C. Bayılmış, “Enabling secure data transmission for wireless sensor networks based iot applications,” *Ain Shams Engineering Journal*, vol. 14, no. 2, p. 101866, 2023.

- [231] L. Xu and F. Wu, “Cryptanalysis and improvement of a user authentication scheme preserving uniqueness and anonymity for connected health care,” *Journal of medical systems*, vol. 39, no. 2, pp. 1–9, 2015.
- [232] K. Sarmila and S. Manisekaran, “Honey encryption and AES based data protection against brute force attack,” in *2022 Sixth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*. IEEE, 2022, pp. 187–190.
- [233] Z. Lu, “Analysis on AES encryption standard and safety,” in *Third International Symposium on Computer Engineering and Intelligent Communications (ISCEIC 2022)*, vol. 12462. SPIE, 2023, pp. 292–297.
- [234] M. Bakro, S. K. Bisoy, A. K. Patel, and M. A. Naal, “Hybrid blockchain-enabled security in cloud storage infrastructure using ECC and AES algorithms,” in *Blockchain based Internet of Things*. Springer, 2022, pp. 139–170.
- [235] Z. Lu and H. Mohamed, “A complex encryption system design implemented by AES,” *Journal of Information Security*, vol. 12, no. 2, pp. 177–187, 2021.
- [236] A. Dunkels, B. Gronvall, and T. Voigt, “Contiki-a lightweight and flexible operating system for tiny networked sensors,” in *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*. IEEE, 2004, pp. 455–462.

- [237] O. Hahm, E. Baccelli, H. Petersen, and N. Tsiftes, “Operating systems for low-end devices in the internet of things: a survey,” *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 720–734, 2015.
- [238] Y. B. Zikria, M. K. Afzal, F. Ishmanov, S. W. Kim, and H. Yu, “A survey on routing protocols supported by the contiki internet of things operating system,” *Future Generation Computer Systems*, vol. 82, pp. 200–219, 2018.
- [239] K. Kritsis, G. Z. Papadopoulos, A. Gallais, P. Chatzimisios, and F. Theoleyre, “A tutorial on performance evaluation and validation methodology for low-power and lossy networks,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 1799–1825, 2018.

VITAE

- Name: Ibraheem Mohammed Ahmed Al-Hejri
- Nationality: Yemeni
- Date of Birth: 01/07/1987
- Email: *alhejri87@gmail.com*
- Permenant Address: Yemen

Education Background:

Ibraheem Al-Hejri received the B.S. degree in computer science from Thamar University, Thamar, Yemen, in 2009, and the M.Sc. degree in Security & Information Assurance from the King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia, in 2018. He got the opportunity to pursue his Ph.D.degree at KFUPM. He completed his Ph.D. in Computer Science (Security & Net-centric Computing) in December 2023. His current research interests include Cybersecurity, Cryptography and Blockchain applications, IoT Security, Wireless Sensor Networks, and Neural Networks. He has several publications in various

international journals and conferences. He served as an anonymous reviewer for technical journals including IEEE Access, Arabian Journal for Science and Engineering (AJSE), Information Systems Security, Computers (An Open Access Journal from MDPI), and SN computer science.